# Cisco IP Solution Center L2VPN and Carrier Ethernet User Guide, 6.0

**Americas Headquarters**
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

Text Part Number: OL-21636-01

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

*Cisco IP Solution Center L2VPN and Carrier Ethernet User Guide, 6.0*

# CONTENTS

# About This Guide

This preface contains the following sections:

- Objective, page xi
- Audience, page xi
- Organization, page xi
- Related Documentation, page xii
- Obtaining Documentation and Submitting a Service Request, page xiv

## Objective

The *Cisco IP Solution Center L2VPN and Carrier Ethernet User Guide, 6.0* contains information about creating L2VPN or VPLS policies and creating and deploying L2VPN or VPLS services using the Cisco IP Solution Center (ISC).

## Audience

This guide is designed for service provider network managers and operators who are responsible for provisioning L2VPN or VPLS for their customers. Network managers and operators should be familiar with the following topics:

- Basic concepts and terminology used in internetworking.
- Layer 2 Virtual Private Network (L2VPN), Virtual Private LAN Service (VPLS), VPN, Multiprotocol Label Switching (MPLS), and terms and technology.
- Network topologies and protocols.

## Organization

This guide is organized as follows:

- Chapter 1, "Getting Started," provides information on getting started tasks for using the L2VPN component of the Cisco IP Solution Center (ISC).
- Chapter 2, "Setting Up the ISC Services," provides information on setting up the ISC service.

- Chapter 3, "Creating a FlexUNI/EVC Ethernet Policy," provides information on creating a FlexUNI/EVC policy.
- Chapter 4, "Managing a FlexUNI/EVC Ethernet Service Request," provides information on creating, deploying, monitoring, and saving FlexUNI/EVC service requests.
- Chapter 5, "Creating a FlexUNI/EVC ATM-Ethernet Interworking Policy," provides information on creating a FlexUNI/EVC ATM-Ethernet Interworking policy.
- Chapter 6, "Managing a FlexUNI/EVC ATM-Ethernet Interworking Service Request," provides information on creating, deploying, monitoring, and saving FlexUNI/EVC ATM-Ethernet Interworking service requests.
- Chapter 7, "Creating an L2VPN Policy," provides information on creating an L2VPN policy.
- Chapter 8, "Managing an L2VPN Service Request," provides information on creating, deploying, monitoring, and saving a L2VPN service requests.
- Chapter 9, "Creating a VPLS Policy," provides information on creating a VPLS policy.
- Chapter 10, "Managing a VPLS Service Request," provides information on creating, deploying, monitoring, and saving VPLS service requests.
- Chapter 11, "Deploying, Monitoring, and Auditing Service Requests," provides information on how to deploy, manage and audit service requests, and how to access task logs.
- Chapter 12, "Using Autodiscovery for L2 Services," provides an overview of L2 service discovery.
- Chapter 13, "Generating L2 and VPLS Reports," provides information on how to set up, run, and format L2 and VPLS reports.
- Appendix A, "Sample Configlets," provides sample configlets for various network scenarios.
- Appendix B, "Working with Templates and Data Files," provides information about using templates and data files in ISC policies and service requests.
- Appendix C, "Setting Up VLAN Translation," provides information on how to set up VLAN translation for L2VPN ERS services.
- Appendix D, "Terminating an Access Ring on Two N-PEs," describes how to terminate an access ring on two N-PEs.
- Appendix E, "ISC Layer 2 VPN Concepts," provides an overview of the major concepts that structure the ISC L2VPN or VPLS service.
- Index

# Related Documentation

The entire documentation set for Cisco IP Solution Center, can be accessed at:

http://www.cisco.com/en/US/products/sw/netmgtsw/ps4748/tsd_products_support_series_home.html

or at:

http://www.cisco.com/go/isc

**Tip** To copy and paste a two-line URL into the address field of your browser, you must copy and paste each line separately to get the entire URL without a break.

The following documents comprise the ISC documentation set:

### General documentation (in suggested reading order)

- *Cisco IP Solution Center Getting Started and Documentation Guide, 6.0.*

  http://www.cisco.com/en/US/docs/net_mgmt/ip_solution_center/6.0/roadmap/docguide.html
- *Release Notes for Cisco IP Solution Center, 6.0.*

  http://www.cisco.com/en/US/docs/net_mgmt/ip_solution_center/6.0/release/notes/relnotes.html
- *Cisco IP Solution Center Installation Guide, 6.0.*

  http://www.cisco.com/en/US/docs/net_mgmt/ip_solution_center/6.0/installation/guide/installation.html
- *Cisco IP Solution Center Infrastructure Reference, 6.0.*

  http://www.cisco.com/en/US/docs/net_mgmt/ip_solution_center/6.0/infrastructure/reference/guide/infrastructure.html
- *Cisco IP Solution Center System Error Messages, 6.0.*

  http://www.cisco.com/en/US/docs/net_mgmt/ip_solution_center/6.0/system/messages/messages.html
- *Cisco IP Solution Center Third Party and Open Source Copyrights, 6.0.*

  http://www.cisco.com/en/US/docs/net_mgmt/ip_solution_center/6.0/third_party/open_source/ISC_Third_Party_and_Open_Source_Copyrights60.html

### Application and technology documentation (listed alphabetically)

- *Cisco IP Solution Center L2VPN and Carrier Ethernet User Guide, 6.0.*

  http://www.cisco.com/en/US/docs/net_mgmt/ip_solution_center/6.0/l2vpn/user/guide/l2vpn60book.html
- *Cisco IP Solution Center MPLS Diagnostics Expert Failure Scenarios Guide, 6.0.*

  http://www.cisco.com/en/US/docs/net_mgmt/ip_solution_center/6.0/mpls_failure_scenarios/user/guide/mdefs.html
- *Cisco IP Solution Center MPLS Diagnostics Expert User Guide, 6.0.*

  http://www.cisco.com/en/US/docs/net_mgmt/ip_solution_center/6.0/mpls_diagnostics/user/guide/mdeuser.html
- *Cisco IP Solution Center MPLS VPN User Guide, 6.0.*

  http://www.cisco.com/en/US/docs/net_mgmt/ip_solution_center/6.0/mpls_vpn/user/guide/mpls60book.html
- *Cisco IP Solution Center Traffic Engineering Management User Guide, 6.0.*

  http://www.cisco.com/en/US/docs/net_mgmt/ip_solution_center/6.0/traffic_management/user/guide/tem.html

### API Documentation

- *Cisco IP Solution Center API Programmer Guide, 6.0.*

  http://www.cisco.com/en/US/docs/net_mgmt/ip_solution_center/6.0/developer/guide/api_gd.html
- *Cisco IP Solution Center API Programmer Reference, 6.0.*

  http://www.cisco.com/en/US/docs/net_mgmt/ip_solution_center/6.0/developer/reference/xmlapi.zip

**Note** All documentation *might* be upgraded over time. All upgraded documentation will be available at the same URLs specified in this document.

# Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at:

http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

Subscribe to the *What's New in Cisco Product Documentation* as a Really Simple Syndication (RSS) feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service and Cisco currently supports RSS Version 2.0.

CHAPTER 1

# Getting Started

This chapter provides a road map to help you get started using the L2VPN component in ISC 6.0. It contains the following sections:

## Overview

Before you can use the L2VPN component to provision Layer 2 services (L2VPN or VPLS), you must complete several installation and configuration steps, as outlined in this chapter. In addition, you should be familiar with basic concepts for ISC and L2VPN (or VPLS) services. The following sections provide a summary of the key tasks you must accomplish to be able to provision L2VPN or VPLS services using ISC. You can use the information in this chapter as a checklist. Where appropriate, references to other sections in this guide or to other guides in the ISC documentation set are provided. See the referenced documentation for more detailed information. After the basic installation and configuration steps are completed for both ISC and the L2VPN component, see the subsequent chapters of this guide to create and provision L2VPN or VPLS services.

## Installing ISC and Configuring the Network

Before you can use the L2VPN module in ISC to provision L2VPN or VPLS services, you must first install ISC and do the basic network configuration required to support ISC. Details on these steps are provided in the *Cisco IP Solution Center Installation Guide, 6.0*. See that guide for information about ISC installation and general network configuration requirements.

**Note** To use the L2VPN component within ISC, you must purchase and activate the L2VPN license.

# Configuring the Network to Support Layer 2 Services

In addition to basic network configuration required for ISC, you must perform the following network configuration steps to support Layer 2 services. Information on doing these steps is not provided in the ISC documentation. See the documentation for your devices for information on how to perform these steps.

1. Enable MPLS on the core-facing interfaces of the N-PE devices attached to the provider core.
2. Set up /32 loopback addresses on N-PE devices. These loopback addresses should be the termination of the LDP connection(s).
3. Set all Layer 2 devices (switches) to VTP transparent mode. This ensures that none of the switches will operate as VLAN servers and will prevent VLAN information from automatically propagating through the network.

# Setting Up Basic ISC Services

After the basic network configuration tasks are completed to support ISC and L2 services, you use ISC to define elements in the ISC repository, such as providers and regions, customers and sites, devices, VLAN and VC pools, NPCs, and other resources that are necessary to provision L2 services. Detailed steps to perform general ISC tasks are covered in the *Cisco IP Solution Center Infrastructure Reference, 6.0*. You can also find a summary of some important ISC set up tasks in this guide in Chapter 2, "Setting Up the ISC Services." The information below is a checklist of basic ISC services you must set up before provisioning L2 services.

## Setting Up Providers, Customers, and Devices

Perform the following steps to set up providers, customers, and devices in the ISC repository. These are global resources that can be used by all ISC services.

1. **Set up service providers and regions.** The region is important because a single provider could have multiple networks. The region is used as a further level of differentiation to allow for such circumstances. To create a provider and a region, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*. See also Defining a Service Provider and Its Regions, page 2-3.
2. **Set up customers and customer sites.** A customer is a requestor of a VPN service from an ISP. Each customer can own many customer sites. Each customer site belongs to one and only one Customer and can own many CEs. For detailed steps to create customers and sites, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*. See also Defining Customers and Their Sites, page 2-4.
3. **Import or add raw devices.** Every network element that ISC manages must be defined as a device in the ISC repository. An element is any device from which ISC can collect information. In most cases, devices are Cisco IOS routers and switches. You can set up devices in ISC manually, through autodiscovery, or through importing device configuration files. For detailed steps for importing, adding, and collecting configurations for devices, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*. See also Chapter 12, "Using Autodiscovery for L2 Services."
4. **Assign devices roles as PE or CE.** After devices are created in ISC, you must define them as customer (CE) or provider (PE) devices. You do this by editing the device attributes on individual devices or in batch editing through the ISC inventory manager. To set device attributes, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*.

## Setting Up the N-PE Loopback Address

Within ISC, you must set the loopback address on the N-PE device(s). For details about this procedure, see Setting the Loopback Addresses on N-PE Devices, page 2-2.

## Setting Up ISC Resources for L2VPN and VPLS Services

Some ISC resources, such as access domains, VLAN pools, and VC pools are set up to support ISC L2VPN and VPLS services only. Perform the following steps to set up these resources.

1. **Create access domain(s).** For L2VPN and VPLS, you create an access domain if you provision an Ethernet-based service and want ISC to automatically assign a VLAN for the link from the VLAN pool. For each Layer 2 access domain, you need a corresponding access domain object in ISC. During creation, you select all the N-PE devices that are associated with this domain. Later, one VLAN pool can be created for an access domain. For detailed steps to create access domains, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*. See also Creating Access Domains, page 2-4.
2. **Create VLAN pool(s).** A VLAN pool is created for each access domain. For L2VPN and VPLS, you create a VLAN pool so that ISC can assign a VLAN to the links. VLAN ID pools are defined with a starting value and a size. For detailed steps to create VLAN pools, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*. See also Creating VLAN Pools, page 2-5.
3. **Create VC pool(s).**VC ID pools are defined with a starting value and a size of the VC ID pool. A given VC ID pool is not attached to any inventory object (a provider or customer). Create one VC ID pool per network. For detailed steps to create VC pools, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*. See also Creating a VC ID Pool, page 2-7.

## Setting Up NPCs

Before creating an L2VPN or VPLS service request, you must predefine the physical links between CEs and PEs or between U-PEs and N-PEs. The Named Physical Circuit (NPC) represents a link going through a group of physical ports. Thus, more than one logical link can be provisioned on the same NPC. Therefore, the NPC is defined once but used by several L2VPN or VPLS service requests. For detailed steps to create NPCs, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*. See also Creating Named Physical Circuits, page 2-8.

## Setting Up VPNs

You must define VPNs before provisioning L2VPN or VPLS services. In L2VPN, one VPN can be shared by different service types. In VPLS, one VPN is required for each VPLS instance. To define VPNs, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*. See also Defining VPNs, page 2-4.

# Working with L2VPN and VPLS Policies and Service Requests

After you have set up providers, customers, devices, and resources in ISC, you are ready to create L2VPN or VPLS policies, provision service requests (SRs), and deploy the services. After the service requests are deployed you can monitor, audit and run reports on them. All of these tasks are covered in this guide. To accomplish these tasks, perform the following steps.

1. **Review overview information about L2 services concepts.** See Appendix E, "ISC Layer 2 VPN Concepts"
2. **Set up a FlexUNI, L2VPN, or VPLS policy.** See the appropriate chapter, depending on the type of policy you want to create:
   - Chapter 3, "Creating a FlexUNI/EVC Ethernet Policy."
   - Chapter 5, "Creating a FlexUNI/EVC ATM-Ethernet Interworking Policy."
   - Chapter 7, "Creating an L2VPN Policy."
   - Chapter 9, "Creating a VPLS Policy."
3. **Provision the FlexUNI, L2VPN, or VPLS service request.** See the appropriate chapter, depending on the type service request you want to provision:
   - Chapter 4, "Managing a FlexUNI/EVC Ethernet Service Request."
   - Chapter 6, "Managing a FlexUNI/EVC ATM-Ethernet Interworking Service Request."
   - Chapter 8, "Managing an L2VPN Service Request."
   - Chapter 10, "Managing a VPLS Service Request."
4. **Deploy the service request.** See Chapter 11, "Deploying, Monitoring, and Auditing Service Requests."
5. **Check the status of deployed services.** You can use one or more of the following methods:
   - Monitor service requests. See Chapter 11, "Deploying, Monitoring, and Auditing Service Requests."
   - Audit service requests. See Chapter 11, "Deploying, Monitoring, and Auditing Service Requests."
   - Run L2 and VPLS reports. See Chapter 13, "Generating L2 and VPLS Reports."

# A Note on Terminology Conventions

The ISC GUI and this user guide use specific naming conventions for Ethernet services. These align closely with the early MEF conventions. This is expected to be updated in future releases of ISC to conform with current MEF conventions. For reference, the equivalent terms used by the MEF forum are summarized in Table 1-1.

See Layer 2 Terminology Conventions, page E-1 for more information on terminology conventions and how these align with underlying network technologies.

*Table 1-1 Ethernet Service Terminology Mappings*

| Term Used in ISC GUI and This User Guide | Current MEF Equivalent Term |
|---|---|
| **L2VPN over MPLS Core** | |
| Ethernet Wire Service (EWS) | Ethernet Private Line (EPL) |
| Ethernet Relay Service (ERS) | Ethernet Virtual Private Line (EVPL) |
| ATM over MPLS (ATMoMPLS) | — |
| Frame Relay over MPLS (FRoMPLS) | — |
| **VPLS Over MPLS Core** | |
| Ethernet Wire Service (EWS) or<br>Ethernet Multipoint Service (EMS) | Ethernet Private LAN (EP-LAN) |
| Ethernet Relay Service (ERS) or<br>Ethernet Relay Multipoint Service (ERMS) | Ethernet Virtual Private LAN (EVP-LAN) |
| **VPLS over Ethernet Core** | |
| Ethernet Wire Service (EWS) | Ethernet Private LAN (EP-LAN) |
| Ethernet Relay Service (ERS) | Ethernet Virtual Private LAN (EVP-LAN) |

CHAPTER 2

# Setting Up the ISC Services

To create L2VPN, VPLS, and FlexUNI/EVC policies and service requests, you must first define the service-related elements, such as target devices, VPNs, and network links. Normally, you create these elements once.

This chapter contains the basic steps to set up the Cisco IP Solution Center (ISC) services for an L2VPN services. It contains the following sections:

**Note** This chapter presents high-level information on ISC services that are relevant to L2VPN. For more detailed information on setting up these and other basic ISC services, see the *Cisco IP Solution Center Infrastructure Reference, 6.0.*

## Creating Target Devices and Assigning Roles (N-PE or U-PE)

Every network element that ISC manages must be defined as a device in the system. An element is any device from which ISC can collect information. In most cases, devices are Cisco IOS routers that function as N-PE, U-PE, or P. For detailed steps to create devices, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*.

# Configuring Device Settings to Support ISC

Two device settings must be configured to support the use of ISC in the network:

- Switches in the network must be operating in VTP transparent mode.
- Loopback addresses must be set on N-PE devices.

**Note** These are the two minimum device settings required for ISC to function properly in the network. You must, of course, perform other device configuration steps for the proper functioning of the devices in the network.

## Configuring Switches in VTP Transparent Mode

For security reasons, ISC requires VTPs to be configured in transparent mode on all the switches involved in ERS or EWS services before provisioning L2VPN service requests. To set the VTP mode, enter the following Cisco IOS commands:

```
Switch# configure terminal
Switch(config)# vtp mode transparent
```

Enter the following Cisco IOS command to verify that the VTP mode has changed to transparent:

```
Switch# Show vtp status
```

## Setting the Loopback Addresses on N-PE Devices

The loopback address for the N-PE has to be properly configured for an Any Transport over MPLS (AToMPLS) connection. The IP address specified in the loopback interface must be reachable from the remote pairing PE. The label distribution protocol (LDP) tunnels are established between the two loopback interfaces of the PE pair. You set the PE loopback address in the Edit PE device window. Access the Edit PE device window in ISC by doing the following steps.

**Step 1** Choose **Service Inventory > Inventory and Connection Manager.**

**Step 2** Choose PE Devices in the Selection window.

**Step 3** Choose a specific device and click the **Edit** button.

To prevent a wrong loopback address being entered into the system, the loopback IP address field on the GUI is read-only. You choose the loopback address with the help of a separate pop-up window, which you access by clicking the **Select** button. This ensures that you choose only a valid loopback address defined on the device.

To further narrow the search, you can check the **LDPTermination Only** check box and click the **Select** button. This limits the list to the LDP-terminating loopback interface(s).

## Setting Up Devices for IOS XR Support

L2VPN in ISC 6.0 supports devices running Cisco's IOS XR software. IOS XR, a new member of the Cisco IOS family, is a unique self-healing and self-defending operating system designed for always-on operation while scaling system capacity up to 92Tbps. In L2VPN, IOS XR is only supported on Cisco XR12000 and CRS-1 series routers functioning as network provider edge (N-PE) devices.

In L2VPN, the following e-line services are supported for IOS XR:

- Point-to-point ERS with or without a CE.
- Point-to-point EWS with or without a CE.

The following L2VPN features are not supported for IOS XR:

- Standard UNI port on an N-PE running IOS XR. (The attribute **Standard UNI Port** in the Link Attributes window is disabled when the UNI is on an N-PE device running IOS XR.)
- SVI interfaces on N-PEs running IOS XR. (The attribute **N-PE Pseudo-wire On SVI** in the Link Attributes window is disabled for IOS XR devices.)
- Pseudowire tunnel selection. (The attribute **PW Tunnel Selection** in the Link Attributes window is disabled for IOS XR devices.)
- EWS UNI (dot1q tunnel or Q-in-Q) on an N-PE running IOS XR.
- Frame Relay/ATM and VPLS services.

To enable IOS XR support in L2VPN, perform the following steps.

**Step 1** Set the DCPL property Provisioning\Service\l2vpn\platform\CISCO_ROUTER\IosXRConfigType to XML.

Possible values are CLI, CLI_XML, and XML (the default).

**Step 2** Create the device in ISC as an IOS XR device, as follows:

a. Create the Cisco device by choosing **Service Inventory > Inventory and Connection Manager > Devices > Create**. The Create Cisco Device window appears.

b. Set the **OS** attribute, located under Device and Configuration Access Information, to IOS_XR.

**Note** For additional information on setting DCPL properties and creating Cisco devices, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*.

**Step 3** Create and deploy L2VPN service requests, following the procedures in this guide.

Sample configlets for IOS XR devices are provided in Appendix A, "Sample Configlets".

# Defining a Service Provider and Its Regions

You must define the service provider administrative domain before provisioning L2VPN. The provider administrative domain is the administrative domain of an ISP with one BGP autonomous system (AS) number. The network owned by the provider administrative domain is called the backbone network. If an ISP has two AS numbers, you must define it as two provider administrative domains. Each provider administrative domain can own many region objects.

For detailed steps to define the provider administrative domain, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*.

# Defining Customers and Their Sites

You must define customers and their sites before provisioning L2VPN. A customer is a requestor of a VPN service from an ISP. Each customer can own many customer sites. Each customer site belongs to one and only one Customer and can own many CPEs. For detailed steps to create customers, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*.

# Defining VPNs

You must define VPNs before provisioning L2VPN or VPLS. In L2VPN, one VPN can be shared by different service types. In VPLS, one VPN is required for each VPLS instance.

To create a VPN, perform the following steps.

**Step 1** Choose **Service Inventory > Inventory and Connection Manager**.

**Step 2** Click **VPNs** in the left column.

The VPNs window appears.

For detailed steps to create VPNs, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*.

**Note** The VPN in L2VPN is only a name used to group all the L2VPN links. It has no intrinsic meaning as it does for MPLS VPN.

# Creating Access Domains

For L2VPN and VPLS, you create an Access Domain if you provision an Ethernet-based service and want ISC to automatically assign a VLAN for the link from the VLAN pool.

For each Layer 2 access domain, you need a corresponding Access Domain object in ISC. During creation, you select all the N-PE devices that are associated with this domain. Later, one VLAN pool can be created for an Access Domain. This is how N-PEs are automatically assigned a VLAN.

Before you begin, be sure that you:

- Know the name of the access domain that you want to create.
- Have created a service provider to associate with the new access domain.
- Have created a provider region associated with your provider and PE devices.
- Have created PE devices to associate with the new access domain.
- Know the starting value and size of each VLAN to associate with the new access domain.
- Know which VLAN will serve as the management VLAN.

To create an Access Domain, perform the following steps.

**Step 1** Choose **Service Inventory > Inventory and Connection Manager**.

**Step 2** Click **Access Domains** in the left column.

The Access Domains window appears.

The Access Domains window contains the following:

- **Access Domain Name**—Lists the names of access domains. The first character must be a letter. The name can contain letters, numbers, and these punctuation characters: period, underscore, and dash. Limit: 80 characters. You can sort the list by access domain name.
- **Provider Name**—Lists the names of providers. Must begin with a letter. Can contain letters, numbers, and these punctuation characters: period, underscore, and dash. Limited to 80 characters. You can sort the list by provider name.
- From the Access Domains window, you can create, edit, or delete access domains using the following buttons:
  - **Create**—Click to create new access domain. Enabled only if you do not select an access domain.
  - **Edit**—Click to edit the selected access domain (select by checking the corresponding box). Enabled only if you select a a single access domain.
  - **Delete**—Click to delete the selected access domain (select by checking the corresponding box). Enabled only if you select one or more access domains.

# Creating VLAN Pools

For L2VPN and VPLS, you create a VLAN pool so that ISC can assign a VLAN to the links. VLAN ID pools are defined with a starting value and a size of the VLAN pool. A VLAN pool can be attached to an access domain. During the deployment of an Ethernet service, VLAN IDs can be autoallocated from the access domain's pre-existing VLAN pools. When you deploy a new service, ISC changes the status of the VLAN pool from Available to Allocated. Autoallocation gives the service provider tighter control of VLAN ID allocation.

You can also allocate VLAN IDs manually.

**Note** When you are setting a manual VLAN ID on an ISC service, ISC warns you if the VLAN ID is outside the valid range of the defined VLAN pool. If so, ISC does not include the manually defined VLAN ID in the VLAN pool. We recommend that you preset the range of the VLAN pool to include the range of any VLAN IDs that you manually assign.

Create one VLAN pool per access domain. Within that VLAN pool, you can define multiple ranges.

Before you begin, be sure that you:

- Know each VLAN pool start number.
- Know each VLAN pool size.

- Have created an access domain for the VLAN pool. See Creating Access Domains, page 2-4.
- Know the name of the access domain to which each VLAN pool will be allocated.

To have ISC automatically assign a VLAN to the links, perform the following steps.

**Step 1** Choose **Service Inventory**.

**Step 2** Choose **Inventory and Connection Manager**.

**Step 3** Choose **Resource Pools**.

The Resource Pools window appears.

**Step 4** Choose **VLAN** from the drop-down **Pool Type** list.

**Step 5** Click **Create**.

The Create VLAN Pool window appears.

**Step 6** Enter a VLAN Pool Start number.

**Step 7** Enter a VLAN Pool Size number.

**Step 8** If the correct access domain is not showing in the Access Domain field, click **Select** to the right of Access Domain field.

The Access Domain for New VLAN Pool dialog box appears.

If the correct access domain is showing, continue with Step 9.

**a.** Choose an Access Domain Name by clicking the button in the Select column to the left of that Access Domain.

**b.** Click **Select**. The updated Create VLAN Pool window appears.

**Step 9** Click **Save**.

The updated VLAN Resource Pools window appears.

**Note** The pool name is created automatically, using a combination of the provider name and the access domain name.

**Note** The Status field reads "Allocated" if you already filled in the Reserved VLANs information when you created the access domain. If you did not fill in the Reserved VLANs information when you created the access domain, the Status field reads "Available." To allocate a VLAN pool, you must fill in the corresponding VLAN information by editing the access domain. (See Creating Access Domains, page 2-4.) The VLAN pool status automatically sets to "Allocated" on the Resource Pools window when you save your work.

**Step 10** Repeat this procedure for each range you want to define within the VLAN.

# Creating a VC ID Pool

VC ID pools are defined with a starting value and a size of the VC ID pool. A given VC ID pool is not attached to any inventory object (a provider or customer). During deployment of an L2VPN or VPLS service, the VC ID can be autoallocated from the same VC ID pool or you can set it manually.

**Note** When you are setting a manual VC ID on an ISC service, ISC warns you if the VC ID is outside the valid range of the defined VC ID pool. If so, ISC does not include the manually defined VC ID in the VC ID pool. We recommend that you preset the range of the VC ID pool to include the range of any VC IDs that you manually assign.

Create one VC ID pool per network.

In a VPLS instance, all N-PE routers use the same VC ID for establishing emulated Virtual Circuits (VCs). The VC-ID is also called the VPN ID in the context of the VPLS VPN. (Multiple attachment circuits must be joined by the provider core in a VPLS instance. The provider core must simulate a virtual bridge that connects the multiple attachment circuits. To simulate this virtual bridge, all N-PE routers participating in a VPLS instance form emulated VCs among them.)

**Note** VC ID is a 32-bit unique identifier that identifies a circuit/port.

Before you begin, be sure that you have the following information for each VC ID pool you must create:

- The VC Pool start number
- The VC Pool size

Perform the following steps for all L2VPN and VPLS services.

**Step 1** Choose **Service Inventory**.

**Step 2** Choose **Inventory and Connection Manager**.

Choose **Resource Pools**.

The Resource Pools window appears.

**Step 3** Choose **VC ID** from the drop-down **Pool Type** list.

Because this pool is a global pool, it is not associated with any other object.

**Step 4** Click **Create**.

The Create VC ID Pool window appears.

**Step 5** Enter a VC pool start number.

**Step 6** Enter a VC pool size number.

**Step 7** Click **Save**.

The updated VC ID Resource Pools window appears.

# Creating Named Physical Circuits

Before creating an L2VPN or VPLS service request, you must predefine the physical links between CEs and PEs. The Named Physical Circuit (NPC) represents a link going through a group of physical ports. Thus, more than one logical link can be provisioned on the same NPC; therefore, the NPC is defined once but used during several L2VPN or VPLS service request creations.

There are two ways to create the NPC links:

- Through an NPC GUI editor. For details on how to do this, see Creating NPCs Through an NPC GUI Editor, page 2-8.
- Through the autodiscovery process. For details on how to do this, see Creating NPC Links Through the Autodiscovery Process, page 2-10.

An NPC definition must observe the following creation rules:

- An NPC must begin with a CE or an up-link of the device where UNI resides or a Ring.
- An NPC must end with an N-PE or a ring that ends in an N-PE.

If you are inserting NPC information for a link between a CE and UNI, you enter the information as:

- Source Device is the CE device.
- Source Interface is the CE port connecting to UNI.
- Destination Device is the UNI box.
- Destination interface is the UNI port.

If you are inserting NPC information for a CE not present case, you enter the information as:

- Source Device is the UNI box.
- Source Interface is the UP-LINK port, not the UNI port, on the UNI box connecting to the N-PE or another U-PE or PE-AGG.
- Destination Device is the U-PE, PE-AGG, or N-PE.
- Destination Interface is the DOWN-LINK port connecting to the N-PE or another U-PE or PE-AGG.

If you have a single N-PE and no CE (no U-PE and no CE), you do not have to create an NPC since there is no physical link that needs to be presented.

If an NPC involves two or more links (three or more devices), for example, it connects ence11, enpe1, and enpe12, you can construct this NPC as follows:

- Build the link that connects two ends: mlce1 and mlpe4.
- Insert a device (enpe12) to the link you just made.
- Click **Insert Device** to insert the device.

# Creating NPCs Through an NPC GUI Editor

To create NPCs through the NPC GUI editor, perform the following steps.

**Step 1** Choose **Service Inventory > Inventory and Connection Manager > Named Physical Circuits**.

The Named Physical Circuits window appears.

To create a new NPC, you choose a CE as the beginning of the link and a N-PE as the end. If more than two devices are in a link, you can add or insert more devices (or a ring) to the NPC.

**Note** The new device or ring added is always placed after the device selected, while a new device or ring inserted is placed before the device selected.

Each line on the Point-to-Point Editor represents a physical link. Each physical link has five attributes:

- **Source Device**
- **Source Interface**
- **Destination Device** (must be an N-PE)
- **Destination Interface**
- **Ring**

**Note** Before adding or inserting a ring in an NPC, you must create a ring and save it in the repository. To obtain information on creating NPC rings, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*.

**Source Device** is the beginning of the link and **Destination Device** is the end of the link.

**Step 2** Click **Create**.

The Create a Named Physical Circuit window appears.

**Step 3** Click **Add Device**.

The Select a Device window appears.

**Step 4** Choose a CE as the beginning of the link.

**Step 5** Click **Select**.

The device appears in the Create a Named Physical Circuit window.

**Step 6** To insert another device or a ring, click **Insert Device** or **Insert Ring**.

To add another device or ring to the NPC, click **Add Device** or **Add Ring**. For this example, click **Add Device** to add the N-PE.

**Step 7** Choose a PE as the destination device.

**Step 8** Click **Select**.

The device appears.

**Step 9** In the Outgoing Interface column, click **Select outgoing interface**.

A list of interfaces defined for the device appears.

**Step 10** Choose an interface from the list and click **Select.**

**Step 11** Click **Save**.

The Named Physical Circuits window now displays the NPC that you created.

## Creating a Ring-Only NPC

To create an NPC that contains only a ring without specifying a CE, perform the following steps.

**Step 1** Choose **Service Inventory > Inventory and Connection Manager > Named Physical Circuits**.

**Step 2** Click **Create**.

The Create a Named Physical Circuit window appears.

**Step 3** Click **Add Ring**.

The Select NPC Ring window appears.

**Step 4** Choose a ring and click **Select**. The ring appears.

**Step 5** Click the **Select device** link to select the beginning of the ring.

A window appears showing a list of devices.

**Step 6** Choose the device that is the beginning of the ring and click **Select**.

**Step 7** Click the **Select device** link to choose the end of the ring.

**Step 8** Choose the device that is the end of the ring and click **Select**.

**Note** The device that is the end of the ring in a ring-only NPC must be an N-PE.

**Step 9** The Create a Named Physical Circuit window appears showing the Ring-Only NPC.

**Step 10** Click **Save** to save the NPC to the repository.

## Terminating an Access Ring on Two N-PEs

ISC supports device-level redundancy in the service topology to provide a failover in case one access link should drop. This is accomplished through a special use of an NPC ring that allows an access link to terminate at two different N-PE devices. The N-PEs in the ring are connected by a logical link using loopback interfaces on the N-PEs. The redundant link starts from a U-PE device and may, optionally, include PE-AGG devices.

For details on how to implement this in ISC, see Appendix D, “Terminating an Access Ring on Two N-PEs.”

## Creating NPC Links Through the Autodiscovery Process

With autodiscovery, the existing connectivity of network devices can be automatically retrieved and stored in the ISC database. NPCs are further abstracted from the discovered connectivity.

For detailed steps to create NPCs using autodiscovery, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*.

# Creating and Modifying Pseudowire Classes for IOS XR Devices

The pseudowire class feature provides you with the capability to configure various attributes associated with a pseudowire that is deployed as part of an L2VPN service request on IOS XR-capable devices.

**Note** The pseudowire class feature is supported for IOS XR 3.6.1 and higher.

The pseudowire class feature supports configuration of the encapsulation, transport mode, fallback options, and selection of a traffic engineering tunnel down which the pseudowire can be directed. For tunnel selection, you can select the tunnel using the ISC Traffic Engineering Management (TEM) application, if it is being used. Otherwise, you can specify the identifier of a tunnel that is already provisioned within the network. For IOS XR-capable devices, the pseudowire class is a separately defined object in the ISC repository, which can be attached to an L2VPN service policy or service request. The pseudowire class feature is only available for use in L2VPN ERS, EWS and ATM policies and service requests.

This section describes how to create and modify pseudowire classes. For information on how the pseudowire class is associated to a L2VPN policy and used within a service request, see Chapter 7, "Creating an L2VPN Policy" and Chapter 8, "Managing an L2VPN Service Request."

## Creating a Pseudowire Class

To create a pseudowire class, perform the following steps.

**Step 1** Navigate to **Service Inventory > Inventory and Connection Manager**.

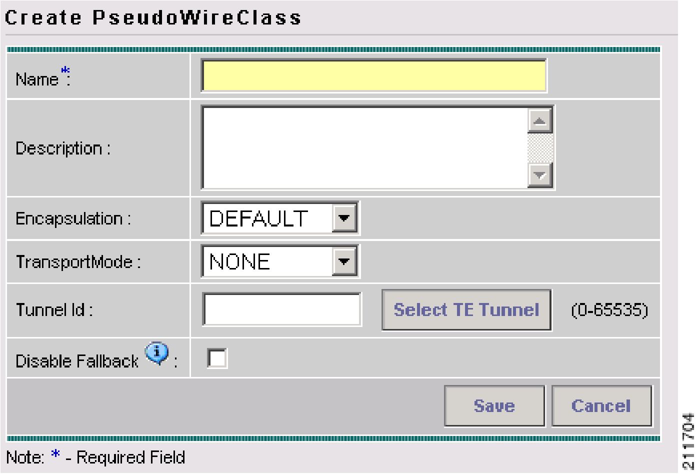
**Step 2** Click the PseudoWireClass icon.

The Pseudowire Classes window appears.

**Step 3** Click the **Create** button.

The Create PseudowireClass window appears, as shown in Figure 2-1.

***Figure 2-1 Create PseudoWireClass Window***



Name*:
Description :
Encapsulation : DEFAULT
TransportMode : NONE
Tunnel Id : (0-65535)
Disable Fallback : □
Note: * - Required Field

**Step 4** In the **Name** field, enter a valid PseudoWireClass name.

The pseudowire class name is used for provisioning **pw-class** commands on the XR device. The name should not exceed 32 characters and should not contain spaces.

**Step 5** In the **Description** field, enter a meaningful description of less than 128 characters.

This field is optional.

**Step 6** Choose the **MPLS** encapsulation type from the **Encapsulation** drop-down list.

**Note** Currently, the only encapsulation type supported is MPLS.

**Step 7** Choose the transport mode from the **TransportMode** drop-down list. The choices are:

- **Ethernet**
- **Vlan**
- **NONE** (default)

**Note** If you want to set the TransportMode to Vlan, we recommend you do this via a pseudowire class, if supported by the version of IOS XR being used. If pseudowire class is not supported in a particular version of IOS XR, then you must set the TransportMode using a Dynamic Component Properties Library (DCPL) property, as explained in the section Configuring the Transport Mode When Pseudowire Classes are Not Supported, page 2-13.

**Step 8** Enter a **Tunnel ID** of a TE tunnel that has already been provisioned by ISC or that has been manually provisioned on the device.

This value is optional. You can also select a TE tunnel that has already been provisioned by ISC, as covered in the next step.

**Step 9** Click **Select TE Tunnel** if you want to select a TE tunnel that has been previously provisioned by ISC.

The Select TE Tunnel pop-up window appears. Choose a TE tunnel and click **Select**. This populates the TE Tunnel field with the ID of the selected TE tunnel.

**Note** After a TE tunnel is associated to a pseudowire class or provisioned in a service request, you will receive an error message if you try to delete the TE tunnel using the Traffic Engineering Management (TEM) application. TE tunnels associated with a pseudowire class or service request cannot be deleted.

**Step 10** Check the **Disable Fallback** check box to disable the fallback option for the pseudowire tunnel.

Choose this option based on your version of IOS XR. It is required for IOS XR 3.6.1 and optional for IOS XR 3.7 and above.

# Modifying a Pseudowire Class Object

This section describes how to modify (edit) an existing pseudowire class and how the editing operation might impact L2VPN service requests.

To modify a pseudowire class, perform the following steps.

**Step 1** Navigate to **Service Inventory > Inventory and Connection Manager > PseudoWireClass**.

The PseudoWire Classes window appears.

**Step 2** Select the pseudowire class object you want to modify, and click **Edit**.

The Edit PseudoWire Class window appears.

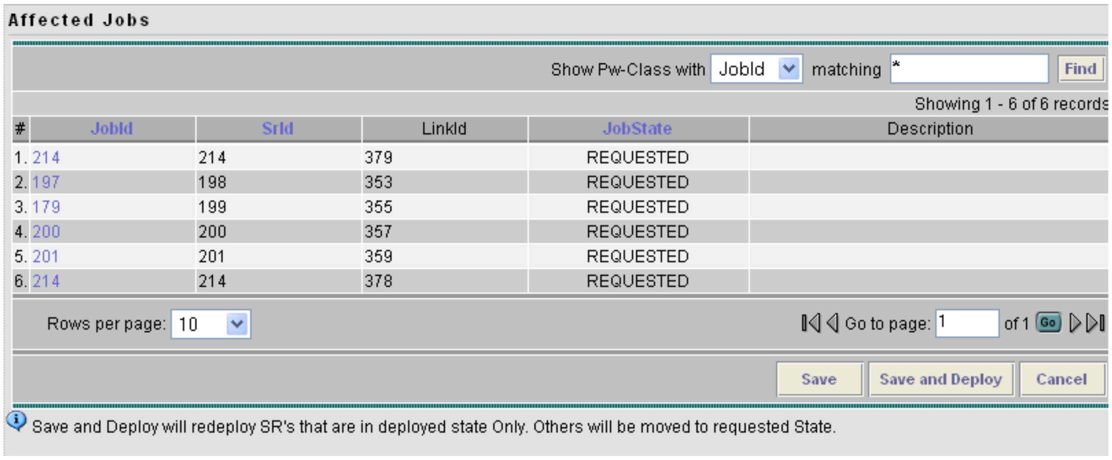
**Step 3** Make the desired changes and click **Save**.

**Note** The Name field is not editable if the pseudowire class is associated with any service requests.

If the pseudowire class being modified is associated with any L2VPN service requests, the Affected Jobs window appears, which displays a list of affected service requests, as shown in Figure 2-2.

**Note** A list of affected service requests only appears if the Transport Mode, Tunnel ID, or Disable Fallback values are changed in the pseudowire class being modified.

*Figure 2-2 Affected Jobs*



Show Pw-Class with JobId matching *

Showing 1 - 6 of 6 records

| # | JobId | SrId | LinkId | JobState | Description |
|---|---|---|---|---|---|
| 1. | 214 | 214 | 379 | REQUESTED | |
| 2. | 197 | 198 | 353 | REQUESTED | |
| 3. | 179 | 199 | 355 | REQUESTED | |
| 4. | 200 | 200 | 357 | REQUESTED | |
| 5. | 201 | 201 | 359 | REQUESTED | |
| 6. | 214 | 214 | 378 | REQUESTED | |

Rows per page: 10

Go to page: 1 of 1

Save and Deploy will redeploy SR's that are in deployed state Only. Others will be moved to requested State.

**Step 4** Click **Save** to update service requests associated with the modified pseudowire class.

The impacted service requests are moved to the Requested state.

**Step 5** Click **Save and Deploy** to update and deploy service requests associated with the modified pseudowire class.

Deployment tasks are created for the impacted service requests that were previously in the Deployed state.

**Step 6** Click **Cancel** to discard changes made to the modified pseudowire class.

In this case, no change of state occurs for any service requests associated with the pseudowire class.

# Configuring the Transport Mode When Pseudowire Classes are Not Supported

This section describes how to configure the pseudowire transport mode to be of type Vlan for versions of IOS XR that do not support pseudowire classes. This is done through setting a Dynamic Component Properties Library (DCPL) property. See the usage notes following the steps for additional information.

Perform the following steps.

**Step 1** In ISC, navigate to **Administration > Control Center > Hosts**.

**Step 2** Check a check box for a specific host and click the **Config** button.

**Step 3** Navigate to the DCPL property **Services\Common\pseudoWireVlanMode**.

**Step 4** Set the property to **true**.

**Step 5** Click **Set Property**.

ISC then generates VLAN transport mode configuration for the pseudowire.

Usage notes:

- To set the transport mode to Vlan, it is recommended that you do this via a pseudowire class, if supported by the version of IOS XR being used. If the pseudowire class feature is not supported, then the transport mode must be set using a DCPL property, as explained in the steps of this section
- The DCPL property pseudoWireVlanMode only sets the default value for PseudoWireClass TransportMode as Vlan if the DCPL property is set to true. Users can always over ride it.
- The DCPL property pseudoWireVlanMode acts in a dual way:
  - It sets a default value for PseudoWireClass TransportMode to Vlan.
  - In the absence of a pseudowire class, it generates a deprecated command **transport-mode vlan**. The **transport-mode vlan** command is a deprecated command in IOS XR 3.6 and later. Thus, when a pseudowire class is selected for an IOS XR device and the DCPL property is also set to true, the **transport-mode vlan** command is not generated. Pseudowire class and the **transport-mode vlan** command do not co-exist. If a pseudowire class is present, it takes precedence over the deprecated **transport-mode vlan** command.
- The value of the DCPL property pseudoWireVlanMode should not be changed during the life of a service request.

# Defining L2VPN Group Names for IOS XR Devices

This section describes how to specify the available L2VPN group names for policies and service requests for IOS XR devices. The choices appear in a drop-down list of the L2VPN Group Name attribute in policies and service requests. The name chosen is used for provisioning the L2VPN group name on IOS XR devices. The choices are defined through setting a Dynamic Component Properties Library (DCPL) property.

Perform the following steps.

**Step 1** In ISC, navigate to **Administration > Control Center > Hosts**.

**Step 2** Check a check box for a specific host and click the **Config** button.

**Step 3** Navigate to the DCPL property **Services\Common\l2vpnGroupNameOptions**.

**Step 4** Enter a comma-separated list of L2VPN group names in the **New Value** field.

**Step 5** Click **Set Property**.

CHAPTER 3

# Creating a FlexUNI/EVC Ethernet Policy

This chapter contains an overview of FlexUNI/EVC support in ISC, as well as the basic steps to create a FlexUNI/EVC Ethernet policy. It contains the following sections:

- Overview of FlexUNI/EVC Support in ISC, page 3-1
- Defining the FlexUNI/EVC Ethernet Policy, page 3-6
- Setting the Service Options, page 3-8
- Setting the FlexUNI Attributes, page 3-10
- Setting the Interface Attributes, page 3-16
- Enabling Template Association, page 3-23

For information on creating FlexUNI/EVC Ethernet service requests, see Chapter 4, "Managing a FlexUNI/EVC Ethernet Service Request."

**Note** For Ethernet (E-Line and E-LAN) services, use of the FlexUNI/EVC policy and service request is recommended. If you are provisioning services using the FlexUNI/EVC syntax, or plan to do so in the future, use the FlexUNI/EVC service. Existing services that have been provisioned using the L2VPN and VPLS service policy types are still supported and can be maintained with those service types. For ATM and FRoMPLS services, use the L2VPN service policy, as before.

## Overview of FlexUNI/EVC Support in ISC

Flexible user network interface (FlexUNI) is a generic approach for creating Ethernet services in ISC. It can, if supported by the hardware, be used for all Ethernet provisioning. (For information on what platforms support FlexUNI/EVC see Platform Support for FlexUNI/EVC in ISC 6.0, page 3-3.) The FlexUNI/EVC policy is flexible and generic and allows for service designers to provide greater service offerings than available through traditional ISC L2VPN and VPLS services.

Certain line cards have interfaces that support the Cisco IOS Ethernet Virtual Circuit (EVC) syntax. These interfaces can be configured with either EVC infrastructure features or with switch-port command-line interface commands (Class). FlexUNI optionally supports the EVC CLI syntax/infrastructure. For this reason, the FlexUNI policies and service requests are referred to by the umbrella term "FlexUNI/EVC." However, it is important to note that FlexUNI/EVC policies and service request are not tied to the new EVC syntax. Service endpoints can use non-EVC syntax also.

Services leveraging the FlexUNI/EVC infrastructure are varied in nature, and there is not always a clear delineation between different services. This is because FlexUNI/EVC provides great flexibility in the way these services can be delivered. This can make it challenging to define the services. For example, a traditional ERS could be delivered in several ways by variations of the Class on the platform.

The FlexUNI/EVC policy and associated service request offer a generic and flexible service construct to support device capabilities. This policy is flexible enough to cater to different service offerings using the EVC architecture. It allows service designers to utilize most of the EVC features in a flexible manner, to match the hardware and platform capabilities.

The FlexUNI/EVC policy can be used to create only a FlexUNI/EVC service request and not any other existing ISC service request types, such as L2VPN, VPLS, and so on. Likewise, a FlexUNI/EVC service request can be created using only a FlexUNI/EVC policy and not any other existing ISC policies.

The FlexUNI/EVC infrastructure provides several benefits to Carrier Ethernet (CE) deployments, including:

- Flexible frame matching.
- Flexible VLAN tag manipulation and/or translation.
- Multiple services on the same port.
- Flexible service mapping.
- VLAN scaling and locally significant VLANs.

FlexUNI/EVC supports a variety of network configurations, such as the following:

- Provisioning of Ethernet access as a EVC-capable EWS interface on the N-PE.
- Interconnecting Ethernet accesses terminating on a single Cisco 7600 N-PE on one or multiple ports in a bridge domain.
- Interconnecting Ethernet accesses terminating on multiple Cisco 7600 N-PEs in a VPLS service.
- FlexUNI/EVC service support on Cisco ASR 9000 Series Routers running IOS XR.
- Services that combine the existing services with the Ethernet access, including the ERS/EWS interworking service.
- Provisioning of E-Line services, in which one or both N-PE interfaces are FlexUNI.

# FlexUNI/EVC Features

This section summarizes the features supported by the FlexUNI/EVC policy and service request in ISC:

- Choice of topology:
  - Customer edge device (CE) directly connected.
  - CE connected through Ethernet access devices.
- Choice of platforms:
  - FlexUNI/EVC on all N-PEs.
  - FlexUNI/EVC on none of the N-PEs.
  - Mix of FlexUNI/EVC and the old infrastructure. This allows both the old and new platforms to co-exist, in order to ensure continued support for deployed platforms.
- Choice of connectivity across the MPLS core (with or without bridge-domain):
  - Pseudo wires

  - VPLS
  - Local (local connects)
- Flexible VLAN handling mechanism that deals with up to two levels of VLAN tags:
  - VLAN matching for service classification. This provides the ability to match both outer and inner VLAN tags, or the ability to match a range of inner VLAN tags.
  - VLAN manipulations, such as pop outer tag, pop inner tag, push outer tag, push inner tag, and VLAN translations (1:1, 2:1, 1:2, 2:2).
- Flexible forwarding options:
  - Configure a pseudowire on the MPLS core directly under a service instance (for E-Line only).
  - Configure a pseudowire on the MPLS core under a switch virtual interface (SVI) by associating it to a bridge domain.

**Note** The appropriate VLAN manipulations are applicable to pseudowire in both cases.

  - Associate traffic from different interfaces and/or VLANs onto a single bridge domain, with appropriate VLAN manipulations for VPLS.
  - Associate traffic from different interfaces and/or VLANs onto a single bridge domain with appropriate VLAN manipulations for local connects.

# Platform Support for FlexUNI/EVC in ISC 6.0

FlexUNI/EVC services are supported on both IOS and IOS XR platforms, as detailed in the following sections.

## IOS Platform Support

The following IOS platforms are supported for the FlexUNI/EVC service:

- IOS 12.2(33)SRB and SRC
- ES20 line cards (2x10GE and 20x1GE)
- Shared Port Adaptor (SPA) Interface Processor-400 (SIP-400) line cards, version 2.0 (2x1GE and 5x1GE)

The interfaces on the ES20 and SIP-400 line cards support the IOS EVC syntax.

Two example platform scenarios are covered in the next sections. Note that the UNI characteristics and the FlexUNI capabilities of the N-PE are not inter-dependent.

### Example 1

FlexUNI/EVC service requests allow operators to add links with either a EVC-capable interface and/or the nonEVC-capable interface on the N-PE. For example, an operator can add three links to a FlexUNI/EVC service request (with VPLS connectivity) with the following configurations:

- Link one has a Cisco 67xx interface and an IOS 12.2 (33)SRB image on a Cisco 7600 N-PE.
- Link two has a Cisco 67xx interface and an IOS 12.2 (33)SRB image on a Cisco 7600 N-PE.
- Link three has an ES20-based interface and an IOS 12.2 (33)SRB image on a Cisco 7600 N-PE.

### Example 2

As far as Layer 2 access nodes are concerned, configurations on the UNI/NNI of a U-PE and/or PE-AGG are not influenced by the FlexUNI/EVC capability on the N-PE. However, if a selected named physical circuit (NPC) with N-PE interface is configured with FlexUNI/EVC, it cannot be provisioned for traditional configuration. An error will be generated while saving the service request.

On the other hand, if a selected NPC with N-PE interface is configured without FlexUNI, it cannot be provisioned for FlexUNI configuration. An error will be generated while saving the service request.

For example, if for link one of a FlexUNI/EVC service request, if the encapsulation is selected as dot1Q, the interface can share other L2 ERS/VPLS ERMS UNIs on the same U-PE/PE-AGG.

If the N-PE interface that is part of the NPC being picked is already configured with non-FlexUNI/EVC features (using an existing L2VPN or VPLS service request), you cannot configure FlexUNI/EVC on it.

**Note** If “Dot1Q Tunnel” is selected as the encapsulation type, the port cannot be shared with other services.

## IOS XR Platform Support

FlexUNI/EVC services are supported on Cisco ASR 9000 Series Routers running IOS XR 3.7.3 and 3.9.0.

The following FlexUNI/EVC features are supported on IOS XR platforms:

- E-line connections. If an ASR 9000 is added on a direct link, only DOT1Q encapsulation is supported for E-Line services. When using L2 access nodes with NPCs, all supported encapsulations are available.
- E-LAN connections.
- Flexible frame matching.
- Flexible VLAN tag manipulation/translation.
- VLAN scaling and locally significant VLANs.
- The ability to create L2 and L3 services under the same physical interface, restricted only to subinterface.
- All the Layer 2 ports on Cisco ASR 9000 devices are trunk ports, hence only trunk port-based configuration is supported.

The following FlexUNI/EVC services are not supported on IOS XR platforms:

- The N-PE Pseudo-wire on SVI attribute is not supported. SVI interfaces are not available on the devices, which restricts support for Standard UNI and Port Security configuration when the UNI is configured on an N-PE.
- xconnect commands are not directly supported under interface configuration. Support for these commands has been moved to different a hierarchy in IOS XR.
- When the UNI is configured on an N-PE device, EWS service is not supported. Since the Cisco ASR 9000 device is a router, all the Layer 2 ports are default trunk. There is no option for configuring access ports, which restricts support for access port-based services.

**Note** Unless otherwise noted, all of the FlexUNI/EVC policy and service request features documented in this guide are applicable for both IOS and IOS XR platforms.

# Device Roles with FlexUNI/EVC

Presently, ISC has U-PE, PE-AGG and N-PE devices. The basic PE device role association of ISC continues for FlexUNI/EVC policy and service requests. In this release of ISC, there are no changes made to the PE role assignment. A device having FlexUNI/EVC capabilities will not call for a change in the existing role assignment in ISC. However, FlexUNI/EVC capabilities in ISC are supported only for interfaces on N-PE and not on PE-AGG or U-PE devices.

**Note** ISC does not support customer edge devices (CEs) for FlexUNI/EVC. If the access port contains any DSLAMS, non-Cisco Ethernet devices and/or other Cisco devices that are not supported by ISC, such nodes and beyond are not in the scope of ISC. In such cases, from the ISC perspective, the interface on the first ISC-managed device is the UNI.

# Topology Overview for FlexUNI/EVC

This section provides examples of various topologies supported with FlexUNI/EVC. As mentioned in the note at the end of section Device Roles with FlexUNI/EVC, page 3-5, ISC does not support customer edge devices (CEs) with FlexUNI/EVC. References to the term “CE” in the following topology variations (such as “CE directly connected” and so on) is only to indicate how the customer or third-party devices connect to the N-PE. For all the cases involving FlexUNI/EVC, the CE is not supported in ISC. Also, any provider device that is not supported by ISC, and which is used in the access circuit, marks the boundary for the scope of ISC, beyond which no devices (that is, towards the CE, and including the unsupported node) is managed by ISC.

## CE Directly Connected and FlexUNI

With this combination, the UNI is the interface on a supported line card, with EVC capability configured. ISC does not configure ISC’s standard UNI functionality (for example, port-security, storm control, and Layer 2 Protocol Tunneling). This is because of lack of command support on the FlexUNI/EVC-capable hardware. Operators can use templates to configure relevant platform supported parameters to realize any of these features not provided by ISC. ISC configures only the service instance with VLAN manipulations and pseudowire, VPLS, or local-connect on the UNI. NPCs are not needed while creating such links because NPCs are only required when there are access nodes between the N-PE and CE. Other intermediate Ethernet access nodes are not involved in this topology.

## CE Directly Connected and No FlexUNI

This is similar to the UNI on N-PE case in ISC. The FlexUNI/EVC service request can be used to create such links with older Cisco 7600 platforms (that is, N-PE interfaces without FlexUNI/EVC capability), but with plans of adding one or more future links with EVC support. If not, one could use the existing ERS/EWS/ERMS/EMS functionality in ISC. NPCs are not needed while creating such links because NPCs are only required when there are access nodes between the N-PE and CE. Other intermediate Ethernet access nodes are not involved in this topology.

## CE Not Directly Connected and FlexUNI

This topology involves the following configurations:

- UNI on a U-PE or PE-AGG to which the CE is connected.

- Ethernet U-PE and/or PE-AGGs.
- N-PE with FlexUNI-capable interface on the CE-facing side.

All service-specific parameters, such as port-security, L2 Protocol Tunneling, storm control, and so on, are applicable to the UNI (Standard UNI) in such links. The U-PE and/or PE-AGG configurations will also have no change in CLIs. However, the EVC commands are applicable only on the N-PE (on the CE-facing interface). NPCs are used while creating such links.

### CE Not Directly Connected and No FlexUNI

This link is identical to an attachment circuit in existing ISC implementations. This has a standard UNI as in existing ISC services. NPCs are used while creating such links.

## A Note on Checking of Configurations

ISC attempts to provision all configurations generated by a FlexUNI/EVC service request. ISC does not perform any prior checks to verify if the CLIs are compatible with the specific devices being provisioned. This is to ensure flexibility of support for device/platform features, which could change over time. Hence, it is important for the service designer or operator to carefully create the FlexUNI/EVC policies and service requests.

# Defining the FlexUNI/EVC Ethernet Policy

You must define a FlexUNI/EVC Ethernet policy before you can provision a service. A policy can be shared by one or more service requests that have similar service requirements.

A policy is a template of most of the parameters needed to define a FlexUNI/EVC service request. After you define it, a FlexUNI/EVC policy can be used by all the FlexUNI/EVC service requests that share a common set of characteristics. You create a new FlexUNI/EVC policy whenever you create a new type of service or a service with different parameters. FlexUNI/EVC policy creation is normally performed by experienced network engineers.

An Editable check box in for an attribute in the policy gives the network operator the option of making a field editable. If the value is set to editable, the service request creator can change the value(s) of the particular policy attribute. If the value is *not* set to editable, the service request creator cannot change the attribute.

You can also associate Cisco IP Solution Center (ISC) templates and data files with a service request. See Appendix B, “Working with Templates and Data Files,” for more about using templates and data files in service requests.

To define a FlexUNI/EVC Ethernet policy, you start by setting the service type attributes. To do this, perform the following steps.

**Step 1** Choose **Service Design** > **Policies**.

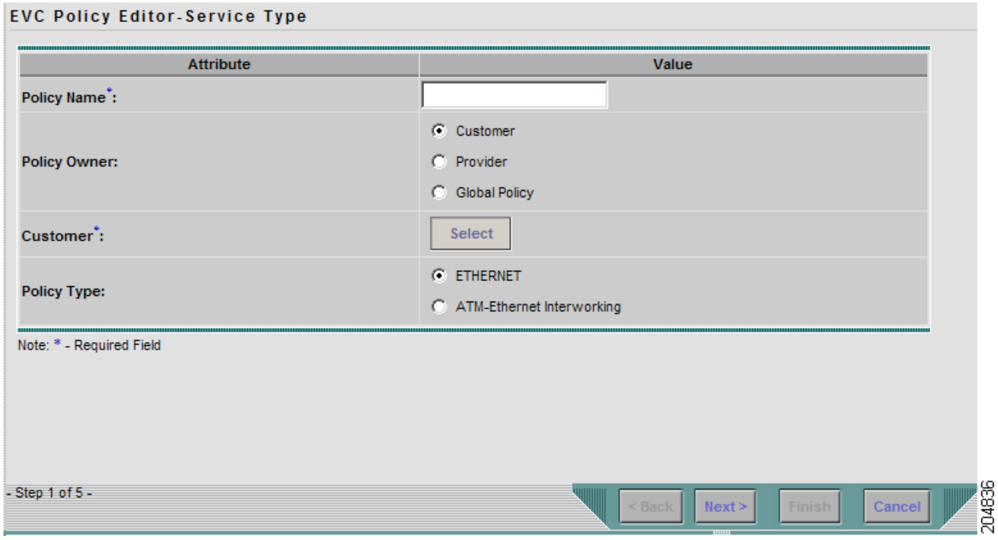
The Policies window appears.

**Step 2** Click **Create**.

**Step 3** Choose **FlexUNI (EVC) Policy**.

The EVC Policy Editor - Service Type window appears, as shown in Figure 3-1.

*Figure 3-1 EVC Policy Editor - Service Type*



EVC Policy Editor-Service Type

| Attribute | Value |
|---|---|
| Policy Name*: | |
| Policy Owner: | ● Customer<br>○ Provider<br>○ Global Policy |
| Customer*: | Select |
| Policy Type: | ● ETHERNET<br>○ ATM-Ethernet Interworking |

Note: * - Required Field

- Step 1 of 5 -

**Step 4** Enter a **Policy Name** for the FlexUNI/EVC policy.

**Step 5** Choose the **Policy Owner** for the FlexUNI/EVC policy.

There are three types of FlexUNI/EVC policy ownership:

- Customer ownership
- Provider ownership
- Global ownership—Any service operator can make use of this policy.

This ownership has relevance when the ISC Role-Based Access Control (RBAC) comes into play. For example, a FlexUNI/EVC policy that is customer-owned can only be seen by operators who are allowed to work on this customer-owned policy. Similarly, operators who are allowed to work on a provider's network can view, use, and deploy a particular provider-owned policy.

**Step 6** Click **Select** to choose the owner of the FlexUNI/EVC policy.

The policy owner was established when you created customers or providers during ISC setup. If the ownership is global, the Select function does not appear.

**Step 7** Choose the **Policy Type**.

The choices are:

- **ETHERNET**
- **ATM-Ethernet Interworking**

**Note** This chapter describes creating the ETHERNET policy type. For information on using the FlexUNI/EVC ATM-Ethernet Interworking policy type, see Chapter 5, "Creating a FlexUNI/EVC ATM-Ethernet Interworking Policy."

**Step 8** Click **Next**.

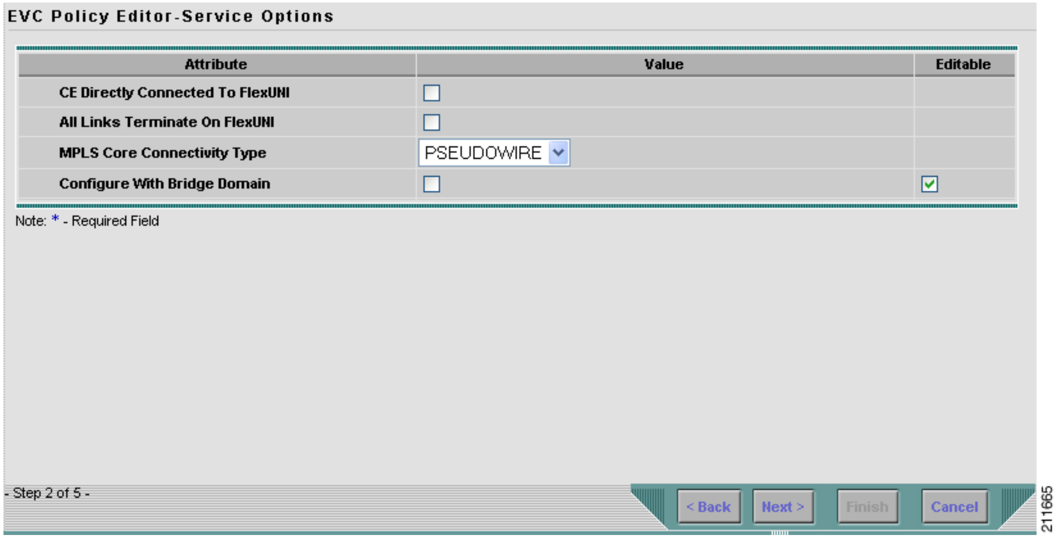
The EVC Policy Editor - Service Options window appears, as show in Figure 3-2.

**Step 9** Continue with the steps contained in the next section, Setting the Service Options, page 3-8.

# Setting the Service Options

This section describes how to set the service options for the FlexUNI/EVC Ethernet policy, as shown in Figure 3-2.

*Figure 3-2 EVC Policy Editor - Service Options Window*



EVC Policy Editor-Service Options

| Attribute | Value | Editable |
|---|---|---|
| CE Directly Connected To FlexUNI | □ | |
| All Links Terminate On FlexUNI | □ | |
| MPLS Core Connectivity Type | PSEUDOWIRE | |
| Configure With Bridge Domain | □ | ☑ |

Note: * - Required Field

- Step 2 of 5 -

The **Editable** check box gives you the option of making a field editable. If you check the **Editable** check box, the service operator who is using this FlexUNI/EVC policy can modify the editable parameter during FlexUNI/EVC service request creation.

To set the FlexUNI/EVC service options, perform the following steps.

**Step 1** Check the **CE Directly Connected to FlexUNI** check box if the CEs are directly connected to the N-PE.

This check box is not checked by default.

Usage notes:

- If the check box is checked, a service request created using this policy can have only directly connected links. No Ethernet access nodes will be involved.
- If the check box is unchecked, a service request created using this policy might or might not have Ethernet access nodes in the links.
- When a CE is directly connected to the N-PE, NPCs are not applicable to the link while creating service requests.
- When a CE is not directly connected to the N-PE, NPCs are used during service request creation, as per standard ISC behavior. There is no change in NPC implementation to support FlexUNI/EVC functionality.

**Step 2** Check the **All Links Terminate on FlexUNI** check box if all links need to be configured with FlexUNI/EVC features.

This check box is not check by default. Usage notes:

- If the check box is checked, a service request created using such policy will have all links using the FlexUNI/EVC feature.

- If the check box is unchecked, zero or more links can use the FlexUNI/EVC feature. This ensures that existing platforms can still be used in one or more links while delivering the services. This allows the possibility of a link with FlexUNI/EVC support being added in the future.

  **Note** If the check box is unchecked, in the service request creation process the user must indicate whether or not the created link is FlexUNI or non-FlexUNI.

- If no links are expected to use the FlexUNI/EVC feature even in the future (for example, if the provider is not planning to upgrade to the EVC infrastructure for the service that is being created), existing ISC policy types (L2VPN or VPLS) can be used instead of FlexUNI/EVC.

**Step 3** Choose an **MPLS Core Connectivity Type** from the drop-down list.

**Note** The core option supports MPLS only. There is no L2TPv3 support for this service.

The choices are:

- **PSEUDOWIRE**—Choose this option to allow connectivity between two N-PEs across the MPLS core. This option does not limit the service to point-to-point (E-Line). This is because even with the PSEUDOWIRE option selected, there can still be multiple CEs connected to a bridge domain on one or both sides of the pseudowire.
- **VPLS**—Choose this option to allow connectivity between multiple N-PEs across the MPLS core.

  There is no limit on the number of N-PEs across the MPLS core within a service request. However, many service requests can refer to the same customer-associated VPN.
- **LOCAL**—Choose this option for local connect cases in which there is no connectivity required across the MPLS core.

  Local connect supports the following scenarios:

  - All interfaces on the N-PE are FlexUNI-capable and using the EVC infrastructure. This is configured by associating all of the customer traffic on these interfaces to a bridge domain. This consumes a VLAN ID on the N-PE (equal to the bridge domain ID).
  - Some interfaces on the N-PE are FlexUNI-capable, while others are switch-port-based. In such cases, all of the customer traffic on the interfaces that are configured with the EVC infrastructure are associated to a bridge domain. The traffic on the non-FlexUNI interfaces (and all the access nodes/interfaces beyond this N-PE) are configured with the Service Provider VLAN ID, where the Service Provider VLAN ID is the same as the bridge domain ID for the EVC-based services.
  - Only two interfaces on the N-PE are involved, and both are based on FlexUNI-capable line cards. In the first case, the operator might choose not to configure the bridge domain option. In this case, the **connect** command that is used for the local connects are used, and the global VLAN is conserved on the device. If the operator chooses to configure with the bridge domain option, both interfaces are associated to a bridge domain ID, so that additional local links can be added to the service in future. This consumes a VLAN ID (bridge domain ID) on the N-PE.

**Note** Attributes available in subsequent windows of the policy workflow dynamically change based on the choice made for the MPLS Core Connectivity Type (PSEUDOWIRE, LOCAL, or VPLS). For completeness, all attributes available for the different core types are documented in the following steps. Attributes apply to all core types, unless otherwise noted.

**Note** Also, some attributes are supported only on IOS or IOS XR platforms. Attributes apply to both platforms, unless otherwise noted. All platform-specific attributes are visible in the policy workflow windows. Later, when a service request is created based on the policy (and specific devices are associated with the service request), platform-specific attributes are filtered from service request windows, depending on the device type (IOS or IOS XR).

**Step 4** Check the **Configure With Bridge Domain** check box to determine bridge domain characteristics.

The behavior of the Configure With Bridge-Domain option works in tandem with the choice you selected in the MPLS Core Connectivity Type option, as follows.

- **PSEUDOWIRE** as the MPLS Core Connectivity Type. There are two cases:

  A. With FlexUNI:

  - If **Configure With Bridge Domain** is checked, the policy configures pseudowires under SVIs associated to the bridge domain.
  - If **Configure With Bridge Domain** is unchecked, the policy will configure pseudowires directly under the service instance. This conserves the global VLAN.

  B. Without FlexUNI:

  - If **Configure With Bridge Domain** is checked, the policy configures pseudowires as in L2VPN services (with SVIs).
  - If **Configure With Bridge Domain** is unchecked, the policy configures pseudowires directly under subinterfaces.

  Only pseudowires can be either configured directly under service instance of the corresponding FlexUNI-capable interface or under SVIs associated to the bridge domain.

- **LOCAL** as the MPLS Core Connectivity Type:
  - If **Configure With Bridge Domain** is checked, the policy allows either point-to-point or multipoint local connect services.
  - If **Configure With Bridge Domain** is unchecked, ISC allows only point-to-point local connects without bridge domain.
- **VPLS—Configure With Bridge Domain** is checked by default and non-editable.

**Step 5** Click **Next**.

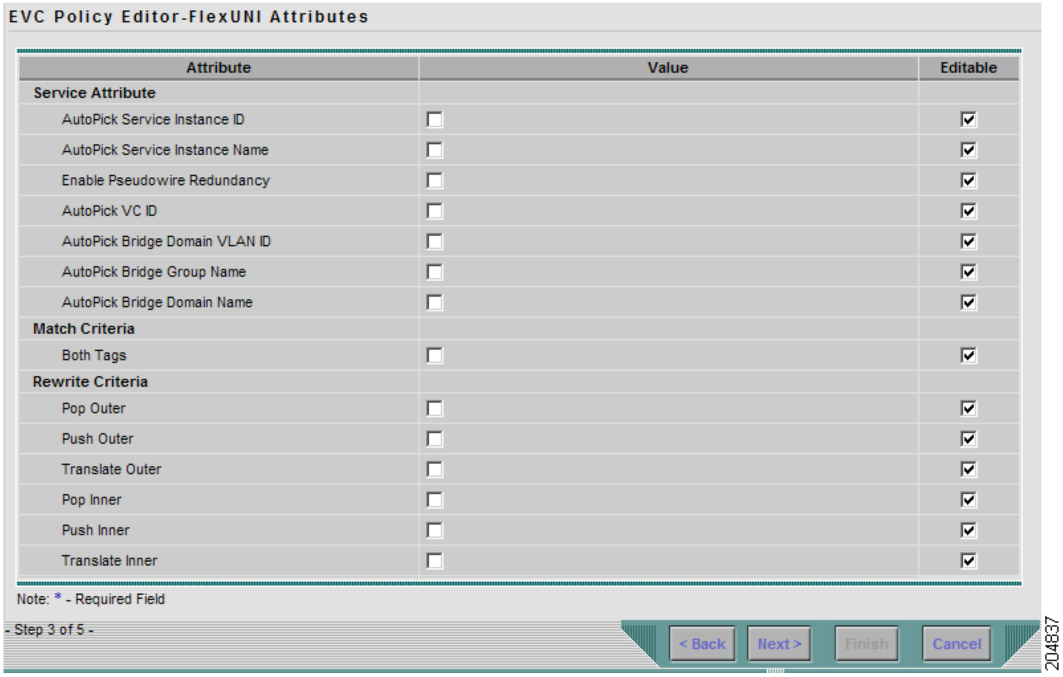
The EVC Policy Editor - FlexUNI Attribute window appears, as shown in Figure 3-3.

**Step 6** Continue with the steps contained in the next section, Setting the FlexUNI Attributes, page 3-10.

# Setting the FlexUNI Attributes

This section describes how to set the FlexUNI attributes for the FlexUNI/EVC Ethernet policy, as shown in Figure 3-3.

*Figure 3-3 EVC Policy Editor - FlexUNI Attribute Window*



EVC Policy Editor-FlexUNI Attributes

| Attribute | Value | Editable |
|---|---|---|
| **Service Attribute** | | |
| AutoPick Service Instance ID | ☐ | ☑ |
| AutoPick Service Instance Name | ☐ | ☑ |
| Enable Pseudowire Redundancy | ☐ | ☑ |
| AutoPick VC ID | ☐ | ☑ |
| AutoPick Bridge Domain VLAN ID | ☐ | ☑ |
| AutoPick Bridge Group Name | ☐ | ☑ |
| AutoPick Bridge Domain Name | ☐ | ☑ |
| **Match Criteria** | | |
| Both Tags | ☐ | ☑ |
| **Rewrite Criteria** | | |
| Pop Outer | ☐ | ☑ |
| Push Outer | ☐ | ☑ |
| Translate Outer | ☐ | ☑ |
| Pop Inner | ☐ | ☑ |
| Push Inner | ☐ | ☑ |
| Translate Inner | ☐ | ☑ |

Note: * - Required Field

- Step 3 of 5 -

FlexUNI attributes are organized under the following categories:.

- Service Attributes
- VLAN Match Criteria
- VLAN Rewrite Criteria

The following sections describe how to set the options under each category.

# Setting the Service Attributes

To set the FlexUNI service attributes, perform the following steps.

**Step 1** Check the **AutoPick Service Instance ID** check box to specify that the service instance ID will be autogenerated and allocated to the link during service request creation.

If the check box is unchecked, while setting the ISC link attributes during service request creation, ISC will prompt the operator to specify the service instance ID.

Usage notes:

- The service instance ID represents an Ethernet Flow Point (EFP) on an interface in the EVC infrastructure. The service instance ID is locally significant to the interface. This ID has to be unique only at the interface level. The ID must be a value from 1 to 8000.
- There are no resource pools available in ISC from which to allocate the service instance IDs.
- It is the responsibility of the operator creating the service request to maintain the uniqueness of the ID at the interface level.

**Step 2** Check the **AutoPick Service Instance Name** check box to have ISC autogenerate a service instance name when you create a service request based on the policy. The autogenerated value is in the following pattern: *CustomerName_ServiceRequestJobID*.

If the check box is unchecked, then you can enter a value during service request creation.

**Step 3** Check the **Enable PseudoWire Redundancy** check box to enable pseudowire redundancy (alternative termination device) under certain conditions.

Usage notes:

- Enable Pseudo Wire Redundancy is only available if the MPLS Core Connectivity Type was set as PSEUDOWIRE in the Service Options window (see Setting the Service Options, page 3-8).
- See Appendix D, "Terminating an Access Ring on Two N-PEs" and, specifically, the section Using N-PE Redundancy in FlexUNI/EVC Service Requests, page D-3, for notes on how this option can be used.

**Step 4** Check the **AutoPick VC ID** check box to have ISC autopick the VC ID during service request creation.

If this check box is unchecked, the operator will be prompted to specify a VC ID during service request creation.

Usage notes:

- This attribute is available only if MPLS Core Connectivity of Type was set as PSEUDOWIRE or VPLS in the Service Options window (see Setting the Service Options, page 3-8).
- When AutoPick VC ID is checked, ISC allocates a VC ID for pseudowires from the ISC-managed VC ID resource pool.
- If MPLS Core Connectivity of Type is VPLS, ISC allocates the VPLS VPN ID from the ISC-managed VC ID resource pool.

**Step 5** Check the **AutoPick Bridge Domain/VLAN ID** check box to have ISC autopick the VLAN ID for the service request during service request creation.

If this check box is unchecked, the operator will be prompted to specify a VLAN ID during service request creation.

Usage notes:

- AutoPick Bridge Domain/VLAN ID consumes a global VLAN ID on the device.
- The bridge domain/VLAN ID is picked from the existing ISC VLAN pool. Once the VLAN ID is assigned in the service request, ISC makes the VLAN ID unavailable for subsequent service requests.
- In the case of manual VLAN ID allocation, ISC does not manage the VLAN ID if the ID lies outside the range of an ISC-managed VLAN pool. In this case, the operator must ensure the uniqueness of the ID in the Ethernet access domain. If an operator specifies a VLAN ID that is within the range of an ISC-managed VLAN pool and the VLAN ID is already in use in the access domain, ISC displays an error message indicating that the VLAN ID is in use.

### Note on Access VLAN IDs

An access VLAN ID is of local significance to the FlexUNI-capable ports. It should not be confused with the global VLANs. This can be visualized as a partitioning of the Ethernet access network beyond the FlexUNI ports into several subEthernet access domains (one each for a FlexUNI-capable port).

However, all the service interfaces on the Ethernet access nodes beyond the FlexUNI ports will have this very same VLAN ID for a link. This ID must be manually specified by the operator when setting the link attributes during service request creation. The operator must ensure the uniqueness of the ID across the FlexUNI-demarcated Ethernet access domain.

These VLAN IDs are not managed by ISC by means of locally-significant VLAN pools. But once a VLAN ID is assigned for a link in the service request, ISC makes the VLAN unavailable for subsequent service requests within the Ethernet access domain demarcated by the FlexUNI. Likewise, if a manually-specified VLAN is already in use in the access domain delimited by the FlexUNI, ISC will display an error message indicating that the new VLAN ID being specified is already in use on the NPC. The operator will be prompted to specify a different VLAN ID, which will be provisioned on the L2 access nodes.

**Step 6** Check the **AutoPick Bridge Group Name** check box to have ISC autopick the group name for the service request during service request creation.

If this check box is unchecked, the operator will be prompted to specify a group name during service request creation. If the check box is checked, the group name will default to the customer name.

**Note** This attribute is applicable only for supported IOS XR devices.

**Step 7** Check the **AutoPick Bridge Domain Name** check box to have ISC autopick the domain name for the service request during service request creation.

Usage notes:

- If this check box is unchecked, the operator will be prompted to specify a domain name during service request creation.
- If the check box is checked, the domain name will default to the following format:
  - For pseudowire and local connect core types: *ISC-Job-Job_ID*, where *Job_ID* is the service request job ID.
  - For VPLS core type: *ISC-VPN_Name-VPN_ID*, where *VPN_Name* is the name of the VPLS VPN being used, and *VPN_ID* is the VPN ID used in the service request.

**Note** This attribute is applicable only for supported IOS XR devices.

**Step 8** Continue with the steps contained in the next section, Setting the VLAN Matching Criteria Attributes, page 3-13.

## Setting the VLAN Matching Criteria Attributes

Prior to the introduction of the FlexUNI capability, service providers could either deploy service-multiplexed services (ERS/ERMS or EVPL/EVCS) or service-bundled services on a single port. Both could not be supported simultaneously due to the limitations in the infrastructure, which only allowed matching the outer-most VLAN tag.

One of the key benefits of FlexUNI/EVC support in ISC is to provide a flexible means to examine the VLAN tags (up to two levels) of the incoming frames and associate them to appropriate Ethernet Flow Points (EFPs). This allows service providers to deploy simultaneously both the service-multiplexed and service-bundled services on a single port.

To set the FlexUNI VLAN matching criteria attributes, perform the following steps.

**Step 1** Check the **Both Tags** check box to enable service requests created with the policy to match both the inner and outer VLAN tags of the incoming frames.

If you do not check this check box, service requests created with the policy will match only the outer VLAN tag of the incoming frames.

Checking the Both Tags attribute causes the Inner VLAN Ranges attribute (covered in the next steps) to appear in the FlexUNI Attribute window.

**Step 2** Check the **Inner VLAN Ranges** check box to enable the range of inner VLAN tags to be specified during service request creation.

If the check box is unchecked, the range of inner VLAN tags are not allowed. In this case, the operator must specify discrete VLAN IDs during service request creation.

**Step 3** Continue with the steps contained in the next section, Setting the VLAN Rewrite Criteria Attributes, page 3-14.

## Setting the VLAN Rewrite Criteria Attributes

Together with VLAN matching criteria, VLAN rewrite makes the FlexUNI/EVC infrastructure very powerful and flexible. The following VLAN rewrite options are supported:

- Pop one or two tags.
- Push one or two tags.
- Translation (1:1, 2:1, 1:2, 2:2).

Be aware of the following considerations when setting the VLAN rewrite criteria attributes:

- Only one kind of rewrite can be done on every CE-facing FlexUNI link.
- All VLAN rewrites are done using the **symmetric** keyword on the ingress traffic (for example, **rewrite ingress tag pop 2 symmetric**).
- For any service instance, only one type of rewrite option (pop, push, or translate) is allowed per instance. For example, if pop out is enabled, push inner, push outer, translate inner, and translate outer are not available.

To set the FlexUNI VLAN rewrite criteria attributes, perform the following steps.

**Step 1** Check the **Pop Outer** check box to pop the outer VLAN ID tag of the incoming frames that fulfill the match criteria.

If this check box is unchecked, the outer tag of the incoming traffic is not popped.

**Step 2** Check the **Pop Inner** check box to pop the inner VLAN ID tag of the incoming frames that fulfill the match-criteria.

If this check box is unchecked, the inner tag is not popped. Note that, if Pop Inner is checked, Pop Outer is automatically checked.

**Step 3** Check the **Push Outer** check box to impose an outer VLAN ID tag onto the incoming frames that fulfill the match criteria.

If this check box is unchecked, no outer tag is imposed on the incoming frames.

Usage notes:

- If Push Outer is checked, all service requests created with the policy push a dot1q outer tag on the incoming frames matching the match criteria. When creating the link during service creation, the operator can specify an outer tag with a value from 1 to 4096.

- This attribute is available regardless of the number of tags used in the match criteria. Whether the incoming traffic is double tagged or single tagged, if Push Outer is enabled, all corresponding service requests push an outer tag. All subsequent nodes consider only the outer-most two tags (if FlexUNI-capable) or just one tag (not FlexUNI-capable) and treat the inner-most tags transparently as payload.
- This VLAN ID is not derived from ISC-managed VLAN ID pools.

**Step 4** Check the **Push Inner** check box to impose an inner VLAN ID tag onto the incoming frames that fulfill the match criteria.

This operation pushes both an inner and an outer tag onto the incoming packet, not just an inner tag. If this check box is unchecked, no inner tag is imposed on the incoming frames.

Usage notes:

- If Push Inner is checked, all service requests created with the policy push a dot1q inner tag on the incoming frames matching the match criteria. When creating the link during service creation, the operator can specify an inner tag with a value from 1 to 4096.
- If Push Inner is checked, Push Outer is automatically checked.
- This attribute is available regardless of the number of tags used in the match criteria. Regardless of whether the incoming traffic is double tagged or single tagged, if Push Inner is enabled, all corresponding service requests push an inner tag. All subsequent nodes consider only the outer-most two tags (if FlexUNI-capable) or just one tag (not FlexUNI-capable) and treat the inner-most tags transparently as payload.
- This VLAN ID is not derived from ISC-managed VLAN ID pools.

**Step 5** Check the **Translate Outer** check box to allow the operator to specify a target outer VLAN ID during service request creation.

The outer tag of all the incoming frames that fulfill the match criteria are translated to this ID. If the check box is unchecked, no outer tag translation is performed. See Table 3-1.

**Step 6** Check the **Translate Inner** check box to allow the operator to specify a target inner VLAN ID during service request creation.

The inner tag of all the incoming frames that fulfill the match criteria are translated to this ID. If the check box is unchecked, no inner tag translation is performed. See Table 3-1.

**Note** Table 3-1 summarizes the realization of different VLAN translations available in the FlexUNI/EVC infrastructure. The second and third columns (Match Outer Tag and Match Inner Tag) refer to policy settings. The last two columns (Translate Outer Tag and Translate Inner Tag) indicate the VLAN translation that occurs on the incoming frames.

*Table 3-1 VLAN Translation Summary Table*

| Type | Match Outer Tag | Match Inner Tag | Translate Outer Tag | Translate Inner Tag |
|---|---|---|---|---|
| 1:1 | True | N/A | Yes | No |
| 1:2 | True | N/A | Yes | Yes |
| 2:1 | True | True | Yes | No |
| 2:2 | True | True | Yes | Yes |

**Step 7** Click **Next**.

The EVC Policy Editor - Interface Attribute window appears, as shown in Figure 3-3.

**Step 8** Continue with the steps contained in the next section, Setting the Interface Attributes, page 3-16.

# Setting the Interface Attributes

This step of creating the FlexUNI/EVC Ethernet policy involves setting the interface attributes, as shown in the EVC Policy Editor - Interface Attribute window in Figure 3-4. The attributes you can configure in this window are grouped under the following categories:

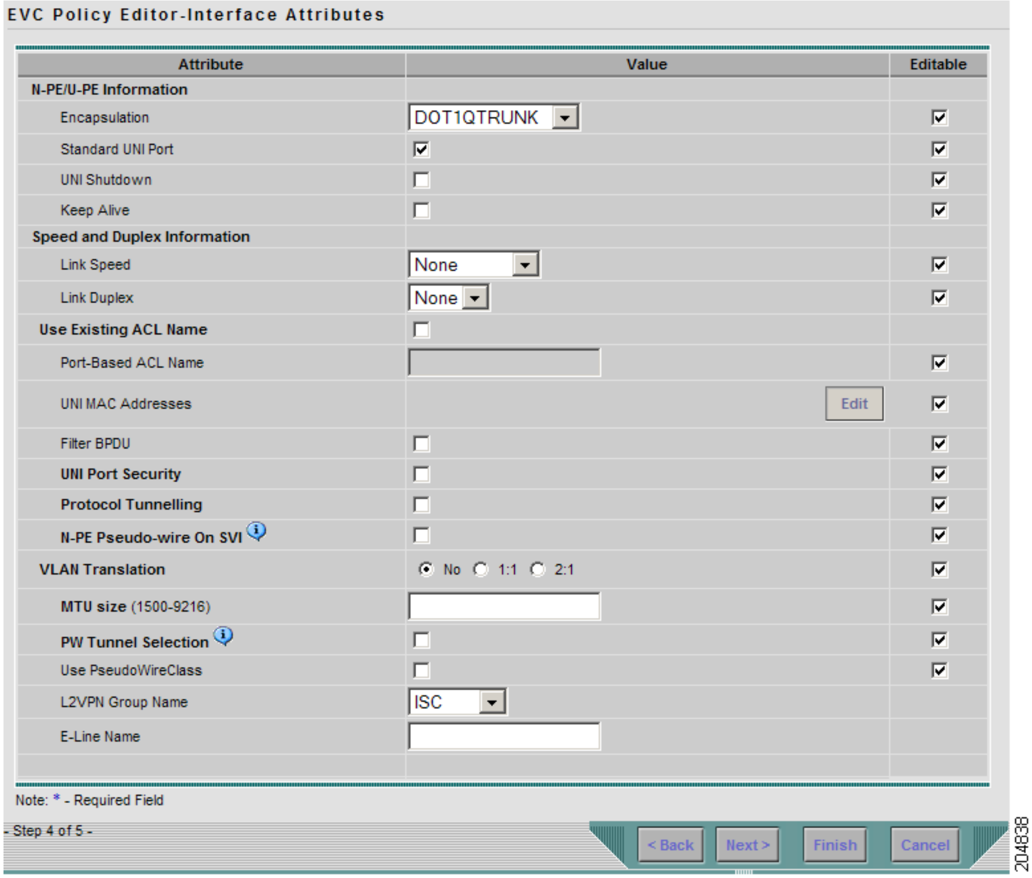
- N-PE/U-PE information
- Speed and duplex information
- ACL name and MAC addresses
- UNI port security
- Storm control
- L2 protocol tunneling

In some cases, checking an attribute causes additional attributes to appear in the GUI. This is covered in the steps that follow.

**Note** If the CE is directly connected to an N-PE, only speed, duplex, UNI shutdown, and other generic options are presented. In this case, port security, storm control, L2 protocol tunneling, and other advanced features are not supported due to the current platform limitations. If these features are needed for a service, the service provider must deploy Layer 2 Ethernet access nodes beyond the FlexUNI to support these requirements.

**Note** Attributes available in the Interface Attributes window dynamically change based on the choice made for the MPLS Core Connectivity Type (PSEUDOWIRE, LOCAL, or VPLS) in the Service Options window (see Setting the Service Options, page 3-8). For completeness, all attributes available for the different core types are documented in the following steps. Attributes apply to all core types, unless otherwise noted.

*Figure 3-4 EVC Policy Editor - Interface Attributes Window*



EVC Policy Editor-Interface Attributes

| Attribute | Value | Editable |
|---|---|---|
| **N-PE/U-PE Information** | | |
| Encapsulation | DOT1QTRUNK | ☑ |
| Standard UNI Port | ☑ | ☑ |
| UNI Shutdown | ☐ | ☑ |
| Keep Alive | ☐ | ☑ |
| **Speed and Duplex Information** | | |
| Link Speed | None | ☑ |
| Link Duplex | None | ☑ |
| **Use Existing ACL Name** | ☐ | |
| Port-Based ACL Name | | ☑ |
| UNI MAC Addresses | Edit | ☑ |
| Filter BPDU | ☐ | ☑ |
| **UNI Port Security** | ☐ | ☑ |
| **Protocol Tunnelling** | ☐ | ☑ |
| **N-PE Pseudo-wire On SVI** | ☐ | ☑ |
| **VLAN Translation** | ● No ○ 1:1 ○ 2:1 | ☑ |
| **MTU size** (1500-9216) | | ☑ |
| **PW Tunnel Selection** | ☐ | ☑ |
| Use PseudoWireClass | ☐ | ☑ |
| L2VPN Group Name | ISC | |
| E-Line Name | | |

Note: * - Required Field

- Step 4 of 5 -

To set the FlexUNI/EVC interface attributes, perform the following steps.

**Step 1** Choose an **Encapsulation** type.

The choices are:

- **DOT1QTRUNK**—Configures the UNI as a trunk with 802.1q encapsulation. If the UNI belongs to a directly connected and FlexUNI link, this setting signifies that the incoming frames are 802.1q encapsulated and that they match the VLAN ID configured for the link. This specific topology does not involve a trunk UNI as such.
- **DOT1QTUNNEL**—Configures the UNI as an 802.1q tunnel (also known as a dot1q tunnel or Q-in-Q) port.
- **ACCESS**—Configures the UNI as an access port.

**Step 2** Check the **Standard UNI Port** check box to enable port security.

This is the default. When you uncheck the check box, the port is treated as an uplink with no security features, and the window dynamically changes to eliminate items related to port security.

**Step 3** Check the **UNI Shutdown** check box if you want to leave the UNI port shut during service activation, for example, when the service provider wants to deploy a service in the network but wants to activate it at a later time.

**Step 4** Check the **Keep Alive** check box to configure keepalives on the UNI port.

By default, this check box is unchecked, which causes the command **no keepalive** to be provisioned on the UNI port. This prevents a CPE from sending keepalive packets to the U-PE, for security purposes. This attribute is editable, in order to support modification on a per-service request basis.

**Step 5** Enter a **Link Speed** (optional) of None, 10, 100, 1000, Auto, or nonegotiate.

**Step 6** Enter a **Link Duplex** (optional) of None, Full, Half, or Auto.

**Step 7** Check the **Use Existing ACL Name** check box if you want to assign your own named access list to the port.

By default, this check box is not checked and ISC automatically assigns a MAC-based ACL on the customer facing UNI port, based on values you enter in **UNI MAC addresses** (below).

**Step 8** Enter a **Port-Based ACL Name** (if you checked the **Use Existing ACL Name** check box, as mentioned in the previous step).

**Note** ISC does not create this ACL automatically. The ACL must already exist on the device, or be added as part of a template, before the service request is deployed. Otherwise, deployment will fail.

**Step 9** Enter one or more Ethernet MAC addresses in **UNI MAC addresses**.

This selection is present only if you uncheck the **Use Existing ACL Name** check box. Click the **Edit** button to bring up a pop-up window in which you enter MAC addresses to be allowed or denied on the port. You can also specify a range of addresses by setting a base MAC address and a filtered MAC address.

**Step 10** Check the **UNI Port Security** check box (see Figure 3-5) if you to want to provision port security-related CLIs to the UNI port by controlling the MAC addresses that are allowed to go through the interface.

a. For **Maximum Number of MAC address**, enter the number of MAC addresses allowed for port security.

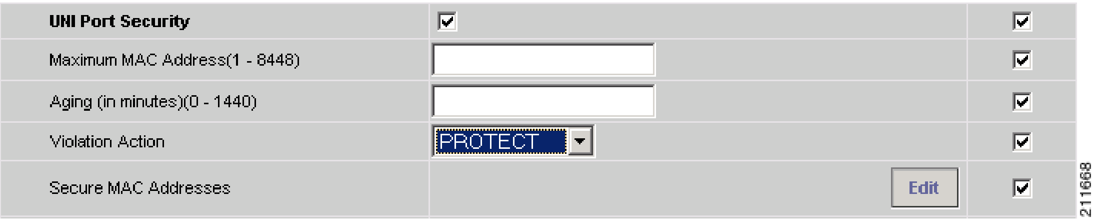
b. For **Aging,** enter the length of time the MAC address can stay on the port security table.

c. For **Violation Action**, choose what action will occur when a port security violation is detected:

- **PROTECT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value.
- **RESTRICT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value and causes the Security Violation counter to increment.
- **SHUTDOWN**—Puts the interface into the error-disabled state immediately and sends an SNMP trap notification.

d. In the **Secure MAC Addresses** field, enter one or more Ethernet MAC addresses.

*Figure 3-5 UNI Port Security*

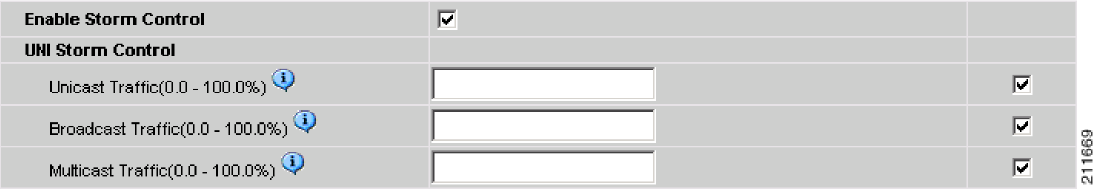


| | | |
|---|---|---|
| **UNI Port Security** | ☑ | ☑ |
| Maximum MAC Address(1 - 8448) | | ☑ |
| Aging (in minutes)(0 - 1440) | | ☑ |
| Violation Action | PROTECT | ☑ |
| Secure MAC Addresses | Edit | ☑ |

**Step 11** Check the **Enable Storm Control** check box (see Figure 3-6) to help prevent the UNI port from being disrupted by a broadcast, multicast, or unicast storm.

Enter a threshold value for each type of traffic. The value, which can be specified to two significant digits, represents the percentage of the total available bandwidth of the port. If the threshold of a traffic type is reached, further traffic of that type is suppressed until the incoming traffic falls below the threshold level.

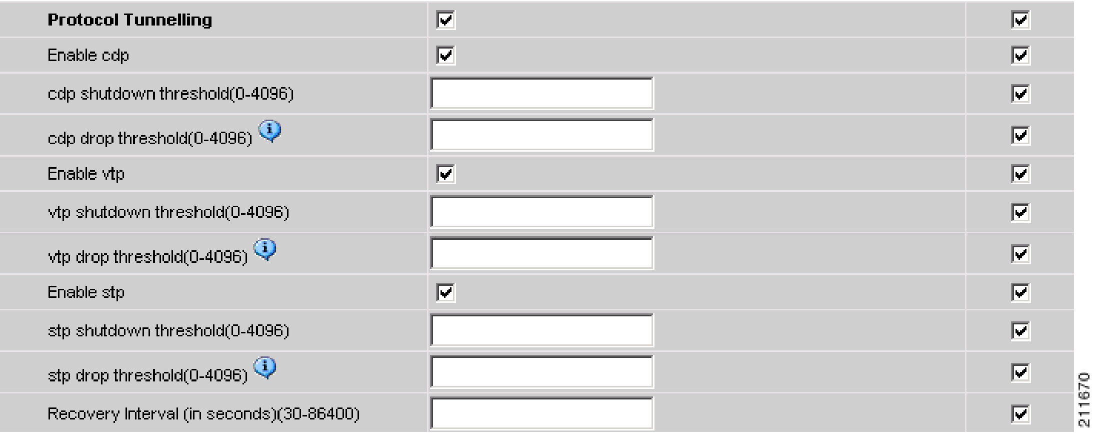
*Figure 3-6 Enable Storm Control*



| | | |
|---|---|---|
| **Enable Storm Control** | ☑ | |
| **UNI Storm Control** | | |
| Unicast Traffic(0.0 - 100.0%) ⓘ | | ☑ |
| Broadcast Traffic(0.0 - 100.0%) ⓘ | | ☑ |
| Multicast Traffic(0.0 - 100.0%) ⓘ | | ☑ |

**Step 12** Check the **Protocol Tunnelling** check box (see Figure 3-7) if you want to define the Layer 2 Bridge Protocol Data Unit (BPDU) frames that can be tunneled over the core to the other end.

*Figure 3-7 Protocol Tunnelling*



| | | |
|---|---|---|
| **Protocol Tunnelling** | ☑ | ☑ |
| Enable cdp | ☑ | ☑ |
| cdp shutdown threshold(0-4096) | | ☑ |
| cdp drop threshold(0-4096) ⓘ | | ☑ |
| Enable vtp | ☑ | ☑ |
| vtp shutdown threshold(0-4096) | | ☑ |
| vtp drop threshold(0-4096) ⓘ | | ☑ |
| Enable stp | ☑ | ☑ |
| stp shutdown threshold(0-4096) | | ☑ |
| stp drop threshold(0-4096) ⓘ | | ☑ |
| Recovery Interval (in seconds)(30-86400) | | ☑ |

For each protocol that you choose, enter the shutdown threshold and drop threshold for that protocol:

a. **Enable cdp**—Enable Layer 2 tunnelling on Cisco Discover Protocol (CDP).

b. **cdp shutdown threshold—**Enter the number of packets per second to be received before the interface is shut down.

c. **cdp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping CDP packets.

d. **Enable vtp**—Enable Layer 2 tunnelling on VLAN Trunk Protocol (VTP).

e. **vtp shutdown threshold**—Enter the number of packets per second to be received before the interface is shut down.

f. **vtp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping VTP packets.

g. **Enable stp—**Enable Layer 2 tunnelling on Spanning Tree Protocol (STP).

h. **stp shutdown threshold—**Enter the number of packets per second to be received before the interface is shut down.

i. **stp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping STP packets.

j. **Recovery Interval**—Enter the amount of time, in seconds, to wait before recovering a UNI port.

**Step 13** Check the **N-PE Pseudo-wire on SVI** check box to have ISC generate forwarding commands under SVIs (switch virtual interfaces).

By default, this check box is not checked. In this case, ISC generates forwarding commands under the service instance.

For a FlexUNI link, the attribute N-PE Pseudo-wire on SVI is dependent on the value of the attribute Configure with Bridge Domain (this is available in the policy workflow in the EVC Policy Editor - Service Options window). N-PE Pseudo-wire on SVI, if enabled, will be reflected only when Configure with Bridge Domain is set to true. Otherwise, the service request will not be created with xconnect under SVI, even if N-PE Pseudo-wire on SVI is enabled.

Usage notes:

- ISC supports a hybrid configuration for FlexUNI/EVC service requests. In a hybrid configuration, the forwarding commands (such as xconnect) for one side of an attachment circuit can be configured under a service instance, and the xconnect configuration for the other side of the attachment circuit can be configured under a switch virtual interface (SVI).
- For examples of these cases, see configlet examples FlexUNI/EVC (Pseudowire Core Connectivity, Bridge Domain, Pseudowire on SVI) and FlexUNI/EVC (Pseudowire Core Connectivity, no Bridge Domain, no Pseudowire on SVI).
- N-PE Pseudo-wire on SVI is applicable for all connectivity types (PSEUDOWIRE, VPLS, and LOCAL), but a hybrid SVI configuration is possible only for pseudowire connectivity.
- When MPLS Core Connectivity Type is set as VPLS, the N-PE Pseudo-wire on SVI attribute is always enabled in the policy and service request.
- When MPLS Core Connectivity Type is set as LOCAL connectivity type, the N-PE Pseudo-wire on SVI attribute is always disabled in the policy and service request.
- The N-PE Pseudo-wire on SVI attribute is not supported for IOS XR devices. Only subinterfaces are supported on ASR 9000 devices; service instance is not supported. All the xconnect commands are configured on L2 subinterfaces.
- Table 3-2 shows various use cases for hybrid configuration for FlexUNI/EVC service requests.

*Table 3-2 Use Cases for Hybrid Configuration for FlexUNI /EVC Service Requests*

| Use Bridge Domain | FlexUNI | N-PE Pseudowire on SVI | CLIs Generated |
|---|---|---|---|
| True | True | True | • xconnect under VLAN interface.<br>• Service instance under main interface. |
| True | True | False | • xconnect under service instance.<br>• Service instance under main interface. |
| False | True | N/A | • xconnect under service instance.<br>• Service instance under main interface. |
| True | False | True | xconnect under VLAN interface. |
| True | False | False | xconnect under subinterface. |
| False | False | False | xconnect under subinterface. |

**Step 14** Specify the type of **VLAN Translation** for this policy by clicking the appropriate radio button.

The choices are:

- **No**—No VLAN translation is performed. (This is the default.)
- **1:1**—1:1 VLAN translation.
- **2:1**—2:1 VLAN translation.

**Note** For detailed coverage of setting up VLAN translation, see Appendix C, "Setting Up VLAN Translation."

**Note** VLAN translation is only supported on links that are specified as non-FlexUNI at the service request level.

**Step 15** Enter the **MTU Size** in bytes.

The maximum transmission unit (MTU) size is configurable and optional. The default size is 9216, and the range is 1500 to 9216. ISC does not perform an integrity check for this customized value. If a service request goes to the Failed Deploy state because this size is not accepted, you must adjust the size until the Service Request is deployed.

In ISC 6.0, different platforms support different ranges.

- For the 3750 and 3550 platforms, the MTU range is 1500 to 1546.
- For the Cisco 7600 Ethernet port, the MTU size is always 9216. Even with the same platform and same IOS release, different line cards support the MTU differently. For example, older line cards only take an MTU size of 9216 and newer cards support 1500 to 9216. However, ISC 6.0 uses 9216 in both cases.
- For the Cisco 7600 SVI (interface VLAN), the MTU size is 1500 to 9216.

**Step 16** Check the **PW Tunnel Selection** check box if you want to be able to manually select the Traffic Engineering (TE) tunnel for the pseudowire connecting point-to-point N-PEs.

This attribute is unchecked by default.

Subsequently, when you create a service request based on this policy, you must specify the TE tunnel ID in a field provided. ISC uses the tunnel information to create and provision a pseudowire class that describes the pseudowire connection between two N-PEs. This pseudowire class can be shared by more than one pseudowire, as long as the pseudowires share the same tunnel ID and remote loopback address. You are responsible to ensure that the tunnel interface and associated ID are configured. During service request creation when you specify the tunnel ID number, ISC does not check the validity of the value. That is, ISC does not verify the existence of the tunnel.

**Step 17** Check the **Use PseudoWireClass** check box to enable the selection of a pseudowire class.

This attribute is unchecked by default.

Usage notes:

- The pseudowire class name is used for provisioning pw-class commands on IOS XR devices. See Creating and Modifying Pseudowire Classes for IOS XR Devices, page 2-10 for additional information on pseudowire class support for IOS XR devices.
- If **Use PseudoWireClass** is checked, an additional attribute, **PseudoWireClass**, appears in the GUI. Click the **Select** button of PseudoWireClass attribute to choose a pseudowire class previously created in ISC.
- The Use PseudoWireClass attribute is only available if the MPLS core connectivity type was set as PSEUDOWIRE in the Service Options window (see Setting the Service Options, page 3-8).
- Use PseudoWireClass is only applicable for IOS XR devices.

**Step 18** For **L2VPN Group Name** choose one of the following from the drop-down list:

- **ISC**
- **VPNSC**

Usage notes:

- This attribute is used for provisioning the L2VPN group name on IOS XR devices.

**Note** The choices in the drop-down list are derived from a configurable DCPL property. For information about how to define the L2VPN Group Name choices available in the drop-down list, see Defining L2VPN Group Names for IOS XR Devices, page 2-14.

- The L2VPN Group Name attribute is not available if the MPLS core connectivity type was set as VPLS in the Service Options window (see Setting the Service Options, page 3-8).
- L2VPN Group Name is only applicable for IOS XR devices.

**Step 19** Enter an **E-Line Name** to specify the point-to-point (p2p) E-line name.

Usage notes:

- If no value is specified for the **E-Line Name** in either the policy or the service request based on the policy, ISC autogenerates a default name as follows:
  - For PSEUDOWIRE core connectivity type, the format is:

    *DeviceName--VC_ID*
  - For LOCAL core connectivity type, the format is:

    *DeviceName--0--VLAN_ID*

  If the default name is more than 32 characters, the device names are truncated.

- The E-Line Name attribute is not available if the MPLS core connectivity type was set as VPLS in the Service Options window (see Setting the Service Options, page 3-8).
- E-Line Name is only applicable for IOS XR devices.

**Step 20** If you would like to enable template association for this policy, click the **Next** button.

See the section Enabling Template Association, page 3-23 for information about this feature.

**Step 21** To save the FlexUNI/EVC policy, click **Finish**.

To create a service request based on a FlexUNI/EVC policy, see Chapter 4, "Managing a FlexUNI/EVC Ethernet Service Request."

# Enabling Template Association

The ISC template feature gives you a means to download free-format CLIs to a device. If you enable templates, you can create templates and data files to download commands that are not currently supported by ISC.

**Step 1** To enable template association for the policy, click the **Next** button in EVC Policy Editor - Interface Attribute window (before clicking **Finish**).

The Template Association window appears. In this window, you can enable template support and, optionally, associate templates and data files with the policy. For instructions about associating templates with policies and how to use the features in this window, see Appendix B, "Working with Templates and Data Files".

**Step 2** When you have completed setting up templates and data files for the policy, click **Finish** in the Template Association window to close it and return to the Policy Editor window.

**Step 3** To save the FlexUNI/EVC policy, click **Finish**.

To create a service request based on a FlexUNI/EVC policy, see Chapter 4, "Managing a FlexUNI/EVC Ethernet Service Request."

CHAPTER 4

# Managing a FlexUNI/EVC Ethernet Service Request

This chapter provides information on how to provision a FlexUNI/EVC Ethernet service request. It contains the following sections:

## Introducing FlexUNI/EVC Service Requests

A FlexUNI/EVC Ethernet service request allows you to configure interfaces on an N-PE to support the FlexUNI/EVC features described in Chapter 3, "Creating a FlexUNI/EVC Ethernet Policy." To create a FlexUNI/EVC service request, a FlexUNI/EVC service policy must already be defined, as described in Chapter 3, "Creating a FlexUNI/EVC Ethernet Policy." Based on the predefined FlexUNI/EVC policy, an operator creates a FlexUNI/EVC service request, with or without modifications to the policy, and deploys the service. One or more templates can also be associated to the N-PE as part of the service request.

Creating a FlexUNI/EVC Ethernet service request involves the following steps:

- Choose an existing FlexUNI/EVC Ethernet policy.
- Choose a VPN.

**Note** When working with VPN objects in the context of FlexUNI/EVC Ethernet policies and service requests, only the VPN name and customer attributes are relevant. Other VPN attributes related to MPLS and VPLS are ignored.

- Specify a bridge domain configuration (if applicable).
- Specify a service request description.
- Specify automatic or manual allocation of the VC ID or VPLS VPN ID.

- Add direct connect links (if applicable).
- Add links with L2 access nodes (if applicable).
- Choose the N-PE and UNI interface for links.
- For links with L2 access nodes, choose a Named Physical Circuit (NPC) if more than one NPC exists from the N-PE or the UNI interface.
- Edit the link attributes.
- Modify the service request.
- Save the service request.

For sample configlets for ATM-Ethernet Interworking scenarios, see Appendix A, “Sample Configlets.”

# Creating a FlexUNI/EVC Service Request

To create a FlexUNI/EVC Ethernet service request, perform the following steps.

**Step 1** Choose **Service Inventory > Inventory and Connection Manager > Service Requests**.

The Service Requests window appears.

**Step 2** Click **Create**.

**Step 3** Choose **FlexUNI (EVC)** from the drop-down list.

The Select EVC Policy window appears. If more than one FlexUNI/EVC policy exists, a list of FlexUNI/EVC policies appears. FlexUNI/EVC service requests must be associated with a FlexUNI/EVC policy. You choose a FlexUNI/EVC policy from the policies previously created (see Chapter 3, “Creating a FlexUNI/EVC Ethernet Policy”).

**Step 4** Choose a FlexUNI/EVC Ethernet policy from the list.

**Step 5** Click **OK**.

The EVC Service Request Editor window appears. The new service request inherits all the properties of the chosen FlexUNI/EVC policy, such as all the editable and non-editable features and pre-set parameters.

**Step 6** Continue with the steps contained in the next section, Setting the Service Request Details, page 4-2.

# Setting the Service Request Details

After you have selected the FlexUNI/EVC Ethernet policy to be used as the basis of the service request, the EVC Service Request Editor window appears. It is divided into three main sections:

- Service Request Details
- Direct Connect Links (no NPCs)
- Links with L2 Access Nodes (involves NPCs)

This window enables you to specify options for the service request, as well as configure directly connected links and links with L2 access nodes. The options displayed in first section of the window change, depending on the MPLS Core Connectivity Type that was specified in the policy (pseudowire, VPLS, or local). For clarity, each of these scenarios is presented in a separate section below, to highlight the different window configurations and behavior of the displayed options.

Proceed to the appropriate section, as determined by the MPLS Core Connectivity Type for the policy:

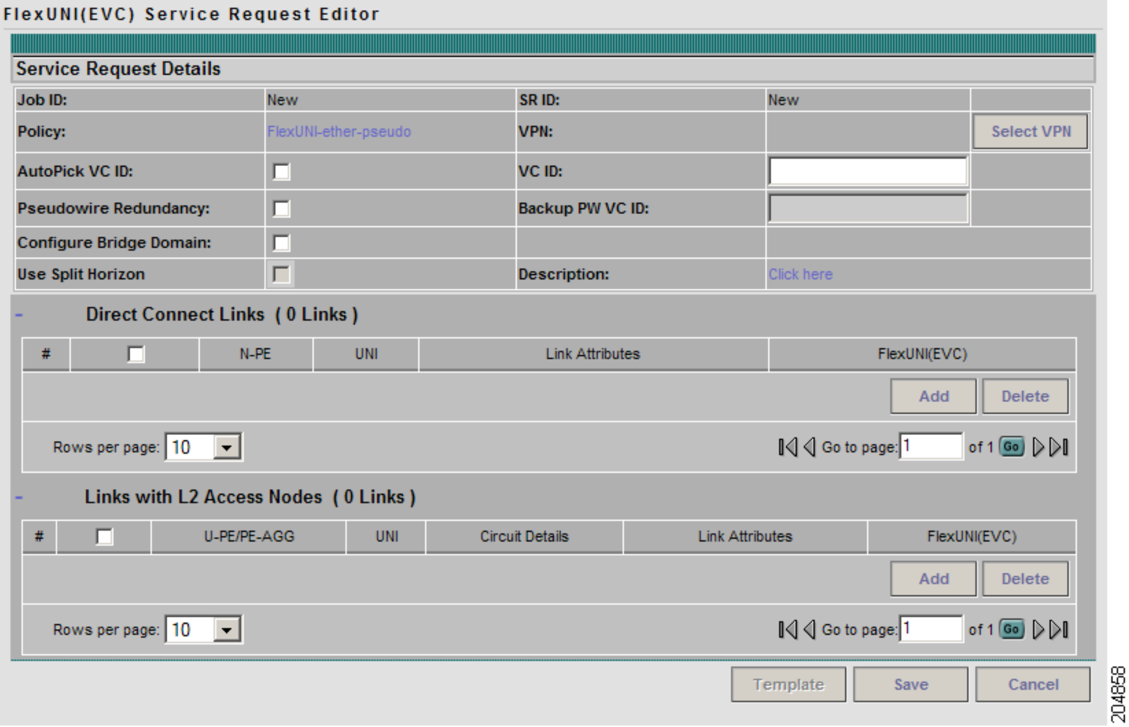
- Pseudowire Core Connectivity, page 4-3
- VPLS Core Connectivity, page 4-5
- Local Core Connectivity, page 4-7

Instructions for setting up direct connect links and links with L2 access nodes are presented in later sections.

## Pseudowire Core Connectivity

If the MPLS Core Connectivity Type for the FlexUNI/EVC Ethernet policy is PSEUDOWIRE, the EVC Service Request Editor window shown Figure 4-1 appears.

*Figure 4-1 EVC Service Request Details Window for Pseudowire Core Connectivity*



FlexUNI(EVC) Service Request Editor

Service Request Details

| | | | | |
|---|---|---|---|---|
| Job ID: | New | SR ID: | New | |
| Policy: | FlexUNI-ether-pseudo | VPN: | | Select VPN |
| AutoPick VC ID: | □ | VC ID: | | |
| Pseudowire Redundancy: | □ | Backup PW VC ID: | | |
| Configure Bridge Domain: | □ | | | |
| Use Split Horizon | □ | Description: | Click here | |

Direct Connect Links ( 0 Links )

| # | □ | N-PE | UNI | Link Attributes | FlexUNI(EVC) |
|---|---|---|---|---|---|

Rows per page: 10 Go to page: 1 of 1

Links with L2 Access Nodes ( 0 Links )

| # | □ | U-PE/PE-AGG | UNI | Circuit Details | Link Attributes | FlexUNI(EVC) |
|---|---|---|---|---|---|---|

Rows per page: 10 Go to page: 1 of 1

Perform the following steps to set the attributes in the first section of the Service Request Details window:

**Note** The **Job ID** and **SR ID** fields are read-only. When the service request is being created for the first time, the fields display a value of NEW. When an existing service request is being modified, the values of the fields indicate the respective IDs that the ISC database holds within the editing flow of the service request.

**Note** The **Policy** field is read-only. It displays the name of the policy on which the service request is based. Clicking on the read-only policy name displays a list of all the attribute values set within the policy.

**Step 1** Click **Select VPN** to choose a VPN for use with this service request.

The Select VPN window appears with the VPNs defined in the system.

> **Note** The same VPN can be used by service requests with LOCAL and PSEUDOWIRE core types. If a VPN for a service request is used with VPLS core type, the same VPN cannot be used for service requests with LOCAL or PSEUDOWIRE core type.

**Step 2** Choose a **VPN Name** in the Select column.

**Step 3** Click **Select**.

The EVC Service Request Editor window appears with the VPN name displayed.

**Step 4** Check the **AutoPick VC ID** check box if you want ISC to choose a VC ID.

If you do not check this check box, you will be prompted to provide the ID in the VC ID field, as covered in the next step.

When AutoPick VC ID is checked, ISC allocates a VC ID for pseudowires from the ISC-managed VC ID resource pool. In this case, the text field for the VC ID option is non-editable.

**Step 5** If AutoPick VC ID was unchecked, enter a VC ID in the **VC ID** field.

Usage notes:

- The VC ID value must be an integer value corresponding to a VC ID.
- When a VC ID is manually allocated, ISC verifies the VC ID to see if it lies within ISC's VC ID pool. If the VC ID is in the pool but not allocated, the VC ID is allocated to the service request. If the VC ID is in the pool and is already in use, ISC prompts you to allocate a different VC ID. If the VC ID lies outside of the ISC VC ID pool, ISC does not perform any verification about whether or not the VC ID allocated. The operator must ensure the VC ID is available.
- The VC ID can be entered only while creating a service. If you are editing the service request, the VC ID field is not editable.

**Step 6** Check the **Enable PseudoWire Redundancy** check box to enable pseudowire redundancy (alternative termination device) under certain conditions.

See Appendix D, "Terminating an Access Ring on Two N-PEs" and, specifically, the section Using N-PE Redundancy in FlexUNI/EVC Service Requests, page D-3, for notes on how this option can be used.

**Step 7** If the AutoPick VC ID attribute was unchecked, enter a VC ID for the backup pseudowire in the **Backup PW VC ID** field.

See the usage notes for the AutoPick VC ID attribute in Step 7, above. The backup VC ID behaves the same as the VC ID of the primary pseudowire.

**Step 8** Check the **Configure Bridge Domain** check box to determine bridge domain characteristics.

The behavior of the Configure Bridge Domain option works in tandem with the choice you selected in the MPLS Core Connectivity Type option in the FlexUNI/EVC policy, which in this case is pseudowire core connectivity. There are two cases:

- With FlexUNI:
  - If **Configure With Bridge Domain** is checked, the policy will configure pseudowires under SVIs associated to the bridge domain.
  - If **Configure With Bridge Domain** is unchecked, the policy will configure pseudowires directly under the service instance. This will conserve the global VLAN.
- Without FlexUNI:
  - If **Configure With Bridge Domain** is checked, the policy will configure pseudowires under SVIs.
  - If **Configure With Bridge Domain** is unchecked, the policy will configure pseudowires directly under subinterfaces.

Pseudowires can be configured either directly under service instance of the corresponding FlexUNI-capable interface or under SVIs associated to the bridge domain.

**Step 9** Check the **Use Split Horizon** check box to enable split horizon with bridge domain.

Usage notes:

- The Use Split Horizon attribute is disabled by default.
- The Use Split Horizon attribute can be used only when the Configure Bridge Domain check box is checked (enabled).
- When Use Split Horizon is enabled, the **bridge domain** command in the CLI will be generated with split horizon. When it is disabled, the **bridge domain** command will be generated without split horizon.

**Step 10** Click the "Click here" link of the **Description** attribute to enter a description label for the service request.

This is useful for searching the ISC database for the particular service request.

A dialogue appears in which you can enter a description.

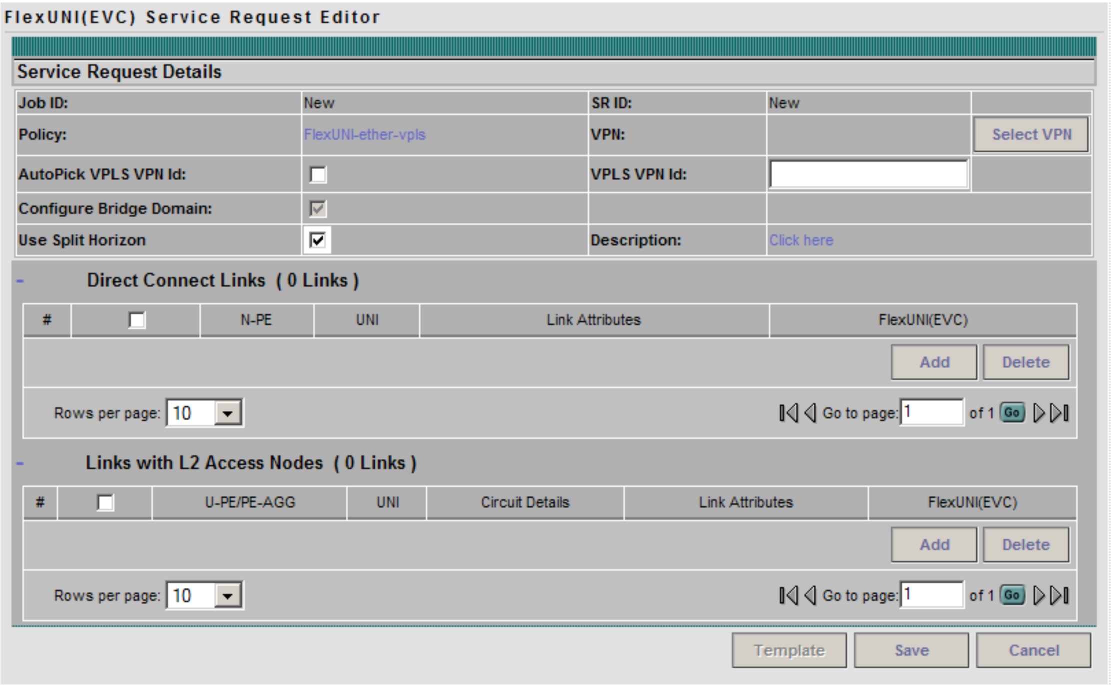
**Step 11** To set up direct connect links, see the section Setting Direct Connect Links, page 4-10.

**Step 12** To set up links with L2 access nodes, see the section Setting Links with L2 Access Nodes, page 4-20.

## VPLS Core Connectivity

If the MPLS Core Connectivity Type for the FlexUNI/EVC Ethernet policy is VPLS, the EVC Service Request Editor window shown Figure 4-2 appears.

*Figure 4-2 EVC Service Request Details Window for VPLS Core Connectivity*



FlexUNI(EVC) Service Request Editor

Service Request Details

| | | | |
|---|---|---|---|
| Job ID: | New | SR ID: | New |
| Policy: | FlexUNI-ether-vpls | VPN: | |
| AutoPick VPLS VPN Id: | □ | VPLS VPN Id: | |
| Configure Bridge Domain: | ☑ | | |
| Use Split Horizon | ☑ | Description: | Click here |

Direct Connect Links ( 0 Links )

| # | □ | N-PE | UNI | Link Attributes | FlexUNI(EVC) |
|---|---|---|---|---|---|

Rows per page: 10 Go to page: 1 of 1

Links with L2 Access Nodes ( 0 Links )

| # | □ | U-PE/PE-AGG | UNI | Circuit Details | Link Attributes | FlexUNI(EVC) |
|---|---|---|---|---|---|---|

Rows per page: 10 Go to page: 1 of 1

Perform the following steps to set the attributes in the first section of the Service Request Details window:

**Step 1** The **Job ID** and **SR ID** fields are read-only.

When the service request is being created for the first time, the fields display a value of NEW. When an existing service request is being modified, the values of the fields indicate the respective IDs that the ISC database holds within the editing flow of the service request.

**Step 2** The **Policy** field is read-only. It displays the name of the policy on which the service request is based.

**Step 3** Click **Select VPN** to choose a VPN for use with this service request.

The Select VPN window appears with the VPNs defined in the system.

**Note** The same VPN can be used by service requests with LOCAL and PSEUDOWIRE core types. If a VPN for a service request is used with VPLS core type, the same VPN cannot be used for service requests with LOCAL or PSEUDOWIRE core type.

**Note** If the same VPN is used among multiple service requests, all having VPLS core type, then all these service requests participate in the same VPLS service.

**Step 4** Choose a **VPN Name** in the Select column.

**Step 5** Click **Select**.

The EVC Service Request Editor window appears with the VPN name displayed.

**Step 6** Check the **AutoPick VPLS VPN ID** check box if you want ISC to choose a VPLS VPN ID.

If you do not check this check box, you will be prompted to provide the VPN ID in the VPLS VPN ID field, as covered in the next step.

- When AutoPick VPLS VPN ID is checked, ISC allocates a VPLS VPN ID from the ISC-managed VC ID resource pool. In this case, the text field for the VPLS VPN ID option is non-editable.
- If AutoPick VPLS VPN ID is checked and a service request already exists that refers to same VPN object, the VPLS VPN ID of the existing service request is allocated to the new service request.

**Step 7** If AutoPick VPLS VPN ID was unchecked, enter a VPLS VPN ID in the **VPLS VPN ID** field.

Usage notes:

- The VPLS VPN ID value must be an integer value corresponding to a VPN ID.
- When a VPLS VPN ID is manually allocated, ISC verifies the VPLS VPN ID to see if it lies within ISC's VC ID pool. If the VPLS VPN ID is in the pool but not allocated, the VPLS VPN ID is allocated to the service request. If the VPLS VPN ID is in the pool and is already in use, ISC prompts you to allocate a different VPLS VPN ID. If the VPLS VPN ID lies outside of the VC ID pool, ISC does not perform any verification about whether the VPLS VPN ID allocated. The operator must ensure the VPLS VPN ID is available.
- The VPLS VPN ID can be entered only while creating a service. If you are editing the service request, the VPLS VPN ID field is not editable.

**Step 8** The **Configure Bridge Domain** check box is check by default and cannot be changed.

Usage notes:

- For VPLS, all configurations are under the SVI.
- When the FlexUNI feature is used, all configurations are under the SVI and also associated to a bridge domain.

**Step 9** Check the **Use Split Horizon** check box to enable split horizon with bridge domain.

Usage notes:

- The Use Split Horizon attribute is disabled by default.
- The Use Split Horizon attribute can be used only when the Configure Bridge Domain check box is checked (enabled).
- When Use Split Horizon is enabled, the **bridge domain** command in the CLI will be generated with split horizon. When it is disabled, the **bridge domain** command will be generated without split horizon.

**Step 10** Click the "Click here" link of the **Description** attribute to enter a description label for the service request.

A dialogue appears in which you can enter a description.

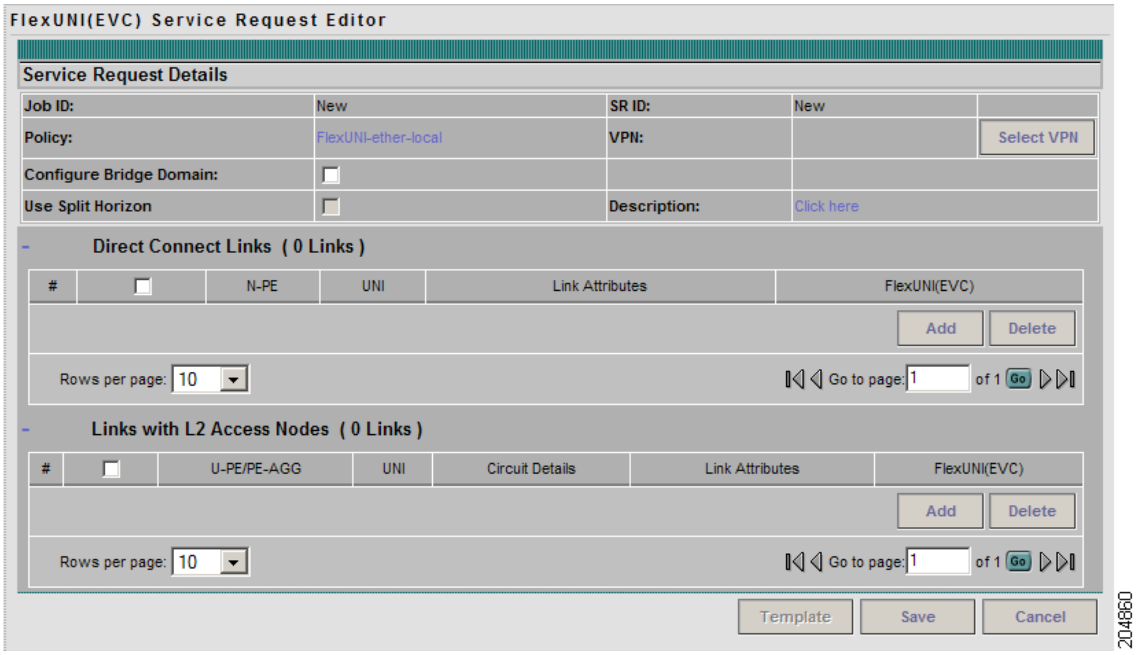
**Step 11** To set up direct connect links, see the section Setting Direct Connect Links, page 4-10.

**Step 12** To set up links with L2 access nodes, see the section Setting Links with L2 Access Nodes, page 4-20.

# Local Core Connectivity

If the MPLS Core Connectivity Type for the FlexUNI/EVC Ethernet policy is LOCAL, the EVC Service Request Editor window shown Figure 4-3 appears.

*Figure 4-3 EVC Service Request Details Window for Local Core Connectivity*



FlexUNI(EVC) Service Request Editor

Service Request Details

| | | | |
|---|---|---|---|
| Job ID: | New | SR ID: | New |
| Policy: | FlexUNI-ether-local | VPN: | |
| Configure Bridge Domain: | ☐ | | |
| Use Split Horizon | ☐ | Description: | Click here |

Direct Connect Links ( 0 Links )

| # | ☐ | N-PE | UNI | Link Attributes | FlexUNI(EVC) |
|---|---|---|---|---|---|

Rows per page: 10

Links with L2 Access Nodes ( 0 Links )

| # | ☐ | U-PE/PE-AGG | UNI | Circuit Details | Link Attributes | FlexUNI(EVC) |
|---|---|---|---|---|---|---|

Rows per page: 10

Perform the following steps to set the attributes in the first section of the Service Request Details window:

**Step 1** The **Job ID** and **SR ID** fields are read-only.

When the service request is being created for the first time, the fields display a value of NEW. When an existing service request is being modified, the values of the fields indicate the respective IDs that the ISC database holds within the editing flow of the service request.

**Step 2** The **Policy** field is read-only.

It displays the name of the policy on which the service request is based.

**Step 3** Click **Select VPN** to choose a VPN for use with this service request.

The Select VPN window appears with the VPNs defined in the system.

**Note** The same VPN can be used by service requests with LOCAL and PSEUDOWIRE core types. If a VPN for a service request is used with VPLS core type, the same VPN cannot be used for service requests with LOCAL or PSEUDOWIRE core type.

**Step 4** Choose a **VPN Name** in the Select column.

**Step 5** Click **Select**.

The EVC Service Request Editor window appears with the VPN name displayed.

**Step 6** Check the **Configure Bridge Domain** check box to determine bridge domain characteristics.

Usage notes:

- If Configure Bridge Domain is checked, all links will have the same bridge domain ID allocated from the VLAN pool on the N-PE. All non-FlexUNI links will have the Service Provider VLAN as the bridge domain ID. On the other hand, if no FlexUNI links are added, the Service Provider VLAN will be allocated first and this will be used as the bridge domain ID when FlexUNI links are added.
- If Configure Bridge Domain is unchecked, a maximum of two links that terminate on the same N-PE can be added. (This uses the **connect** command available in the EVC infrastructure.)

  **Note** See the following comments for details on how ISC autogenerates the connect name.

  Because the device only accepts a maximum of 15 characters for the connect name, the connect name is generated using the following format:

  *CustomerNameTruncatedToMaxPossibleCharacters_ServiceRequestJobID*

  For example, if the customer name is NorthAmericanCustomer and the service request job ID is 56345, the autogenerated connect name would be NorthAmer_56345.

  The CLI generated would be:

  ```
  connect NorthAmer_56345 GigabitEthernet7/0/5 11 GigabitEthernet7/0/4 18
  ```

  In this case, 11 and 18 are service instance IDs.
- If the policy setting for Configure Bridge Domain is non-editable, the option in the service request will be read-only.

**Step 7** Check the **Use Split Horizon** check box to enable split horizon with bridge domain.

Usage notes:

- The Use Split Horizon attribute is disabled by default.
- The Use Split Horizon attribute can be used only when the Configure Bridge Domain check box is checked (enabled).
- When Use Split Horizon is enabled, the **bridge domain** command in the CLI will be generated with split horizon. When it is disabled, the **bridge domain** command will be generated without split horizon.

**Step 8** Click the “Click here” link of the **Description** attribute to enter a description label for the service request.

A dialogue appears in which you can enter a description.

**Step 9** To set up direct connect links, see the section Setting Direct Connect Links, page 4-10.

**Step 10** To set up links with L2 access nodes, see the section Setting Links with L2 Access Nodes, page 4-20.

# Setting up Links to the N-PE

The lower two sections of the EVC Service Request Editor window allow you to set up links to the N-PE. For direct connect links, the CE is directly connected to the N-PE, with no intermediate L2 access nodes. For links with L2 access nodes, there are intermediate devices present between the CE and NPE requiring NPCs to be created in ISC.

The Direct Connect Link section of the window is where you set up links that directly connect to the N-PE. No NPC are involved. The Links with L2 Access Nodes section is where you set up links with L2 (Ethernet) access nodes. NPCs are involved.

See the appropriate section, depending on which type of link you are setting up:

- Setting Direct Connect Links, page 4-10
- Setting Links with L2 Access Nodes, page 4-20

**Note** Many of steps for setting up the two link types are the same. The basic workflow for setting up links, as well as the attributes to be set, are presented in the section Setting Direct Connect Links, page 4-10. Even if you are setting up links with L2 access nodes, it will be helpful to refer to the information presented in that section, as the section on L2 access nodes only covers the unique steps for such links.

## Setting Direct Connect Links

Perform the following steps to set up the direct connect links. Most of these steps apply to links with L2 access nodes also.

**Step 1** Click **Add** to add a link.

A new numbered row for the link attributes appears.

**Step 2** Click **Select NPE** in N-PE column.

The Select PE Device window appears. This window displays the list of currently defined PEs.

a. The **Show PEs with** drop-down list shows PEs by Provider, PE Region Name, or by Device Name.

b. The **Find** button allows a search for a specific PE or a refresh of the window.

c. The **Rows per page** drop-down list allows the user to configure the number of entries displayed on the screen at one time.

**Step 3** In the **Select** column, choose the PE device name for the link.

**Step 4** Click **Select**.

The EVC Service Request Editor window reappears displaying the name of the selected PE in the NPE column.

**Step 5** Choose the UNI interface from the drop-down list in the UNI column.

**Note** ISC only displays the available interfaces for the service, based on the configuration of the underlying interfaces, existing service requests that might be using the interface, and the customer associated with the service request. You can click the **Details** button to display a pop-up window with information on the available interfaces, such as interface name, customer name, VPN name, job ID, service request ID, service request type, translation type, and VLAN ID information.

**Note** When the UNI is configured on an N-PE device running IOS XR, the Standard UNI Port attribute is not supported. All the CLIs related to Standard UNI Port and UNI Port Security are ignored in this case.

**Step 6** Check the **FlexUNI** check box to mark the link for configuring service instance for the links.

**Note** The FlexUNI check box is mentioned at this stage because the setting of the check box alters the behavior of the link editing function available in the Link Attributes column. This is covered in the next steps.

### Editing the Link Attributes

The next steps document the use of the **Edit** link in the Link Attributes column. (In the case where the link attributes have already been set, this link changes from **Edit** to **Change**.) The link editing workflow changes depending on the status of the FlexUNI check box for the link. If the FlexUNI check box is checked, the editing workflow involves setting attributes in two windows, for two sets of link attributes:

- The FlexUNI Details
- Standard UNI Details

If the FlexUNI check box for the link is not checked, only the Standard UNI Details window is presented.

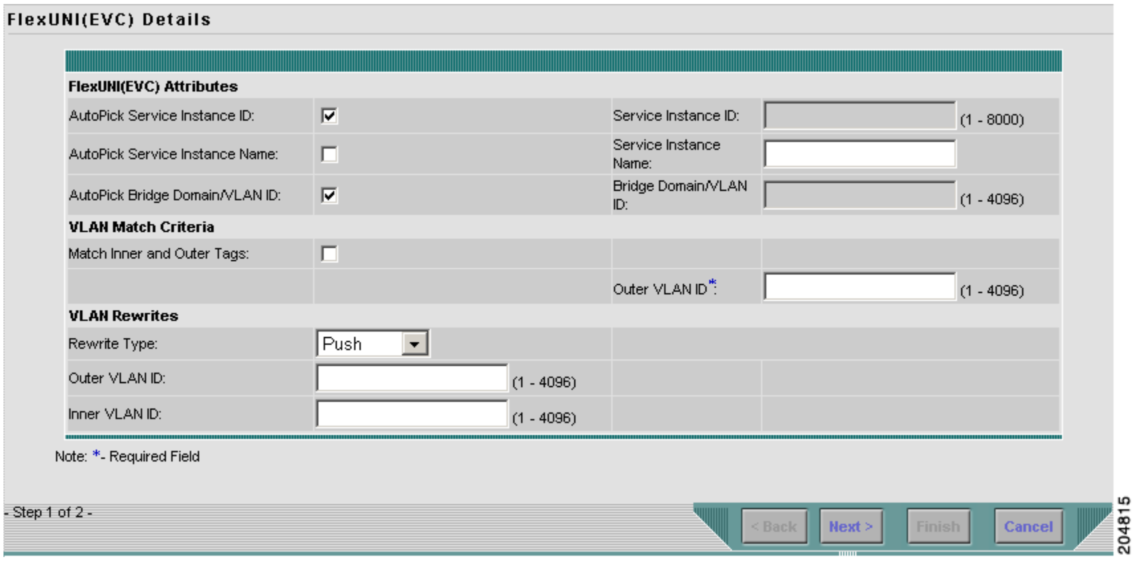
In the steps that follow, both scenarios covered.

**Step 7** Click **Edit** in the Link Attributes column to specify the UNI attributes.

### FlexUNI Details Window

If the FlexUNI check box is checked, the FlexUNI Details window appears, as shown in Figure 4-4.

*Figure 4-4 FlexUNI Details Window*



FlexUNI(EVC) Details

| | | | |
|---|---|---|---|
| **FlexUNI(EVC) Attributes** | | | |
| AutoPick Service Instance ID: | ☑ | Service Instance ID: | (1 - 8000) |
| AutoPick Service Instance Name: | ☐ | Service Instance Name: | |
| AutoPick Bridge Domain/VLAN ID: | ☑ | Bridge Domain/VLAN ID: | (1 - 4096) |
| **VLAN Match Criteria** | | | |
| Match Inner and Outer Tags: | ☐ | | |
| | | Outer VLAN ID*: | (1 - 4096) |
| **VLAN Rewrites** | | | |
| Rewrite Type: | Push | | |
| Outer VLAN ID: | (1 - 4096) | | |
| Inner VLAN ID: | (1 - 4096) | | |

Note: *- Required Field

- Step 1 of 2 -

All of the fields in the FlexUNI Details screen are enabled based on the policy settings. For example, if Both Tags is selected in the policy and is editable, then the Match Inner and Outer Tags check box will be selected and editable in this window. The behavior is similar for the other attributes in the FlexUNI Details window

**Step 8** Check the **AutoPick Service Instance ID** check box to specify that the service instance ID will be autogenerated and allocated to the link during service request creation.

If the check box is unchecked, you must specify the service instance ID (see the next step).

Usage notes:

- The service instance ID represents an Ethernet Flow Point (EFP) on an interface in the EVC infrastructure. The service instance ID is locally significant to the interface. This ID has to be unique only at the interface level. The ID must be a value from 1 to 8000.
- There are no resource pools available in ISC from which to allocate the service instance IDs.

- In the case of a manually provided service instance ID, it is the responsibility of the operator to maintain the uniqueness of the ID at the interface level.
- This attribute is not displayed for IOS XR devices.

**Step 9** If the AutoPick Service Instance ID check box is not checked, enter an appropriate value for the service instance ID in the **Service Instance ID** field.

This attribute is not displayed for IOS XR devices.

**Step 10** Check the **AutoPick Service Instance Name** check box to specify that the service instance name will be autogenerated.

If the check box is unchecked, you can specify the service instance name (see the next step).

Usage notes:

- If the check box is checked, the Service Instance Name text field is disabled.
- The service instance name is autogeneratedi n the following pattern: *CustomerName_ServiceRequestJobID*.
- For example configlets, see FlexUNI/EVC (No AutoPick Service Instance Name, No Service Instance Name), page A-42, FlexUNI/EVC (User-Provided Service Instance Name, Pseudowire Core Connectivity), page A-43, and FlexUNI/EVC (User-Provided Service Instance Name, Local Core Connectivity), page A-44.
- This attribute is not displayed for IOS XR devices.

**Step 11** If the AutoPick Service Instance Name check box is not checked, enter an appropriate value for the service instance ID in the **Service Instance Name** field.

Usage notes:

- The text string representing the service instance name must be 40 characters or less and contain no spaces. Other special characters are allowed.
- If AutoPick Service Instance Name is unchecked and no service instance name is entered in the text field, then ISC does not generate the global **ethernet evc evcname** command in the device configuration generated by the service request.

**Step 12** Check the **AutoPick Bridge Domain/VLAN ID** check box to have ISC autopick the VLAN ID for the service request during service request creation.

If this check box is unchecked, the you must specify a bridge domain VLAN ID (see the next step).

Usage notes:

- AutoPick Bridge Domain/VLAN ID consumes a global VLAN ID on the device.
- The bridge domain VLAN ID is picked from the existing ISC VLAN pool.

**Step 13** If the AutoPick Bridge Domain/VLAN ID check box is unchecked, enter an appropriate value in the **Bridge Domain/VLAN ID** field.

**Note** This configuration applies in conjunction with the Configure Bridge Domain option in the EVC Service Request Editor window. If the option is not enabled in that window, then AutoPick Bridge Domain/VLAN ID check box is redundant and not required.

When a VLAN ID is manually allocated, ISC verifies the VLAN ID to see if it lies within ISC's VLAN ID pool. If the VLAN ID is in the pool but not allocated, the VLAN ID is allocated to the service request. If the VLAN ID is in the pool and is already in use, ISC prompts you to allocate a different VLAN ID. If the VLAN ID lies outside of the ISC VLAN ID pool, ISC does not perform any verification about whether the VLAN ID allocated. The operator must ensure the VLAN ID is available.

**Step 14** Check the **Match Inner and Outer Tags** check box to enable service requests created with the policy to match both the inner and outer VLAN tags of the incoming frames.

If you do not check this check box, service requests created with the policy will match only the outer VLAN tag of the incoming frames.

Checking the Match Inner and Outer Tags attribute causes the Inner VLAN ID and Outer VLAN ID fields (covered in the next steps) to appear.

**Step 15** If the Match Inner and Outer Tags check box is checked, enter the inner and outer VLAN tags in the **Inner VLAN ID** and **Outer VLAN ID** fields.

Usage notes:

- You can specify single values, single ranges, multiples values, multiple ranges, or combinations of these. Examples:
  - 10
  - 10, 15,17
  - 10-15
  - 10-15,17-20
  - 10,20-25
- If the Inner VLAN Ranges attribute is set to true in the policy, the Inner VLAN ID field can take a range of inner VLAN tags.

**Step 16** If the Match Inner and Outer Tags check box is unchecked, enter the outer VLAN tag in the **Outer VLAN ID** field.

**Note** The VLAN specified in Outer VLAN ID will be provisioned on the rest of the L2 access nodes (if the link has any), including the customer-facing UNI.

**Step 17** In the VLAN Rewrite section of the window, choose a **Rewrite Type** from the drop-down list.

The choices are:

- **Pop**
- **Push**
- **Translate**

The subsequent attributes in the GUI change depending on the choice of Rewrite Type, as covered in the next steps.

**Step 18** If Pop is the Rewrite Type, two check boxes appear:

**a.** Check the **Pop Outer Tag** check box to pop the outer VLAN ID tag of the incoming frames that fulfill the match criteria. If this check box is unchecked, the outer tag of the incoming traffic will not be popped.

**b.** Check the **Pop Inner Tag** check box to pop the inner VLAN ID tag of the incoming frames that fulfill the match-criteria. If this check box is unchecked, the inner tag will not be changed.

Note that if Pop Inner Tag is checked, Pop Outer Tag is automatically checked.

**Step 19** If Push is the Rewrite Type, two text boxes appear:

**a.** In the text box **Outer VLAN ID**, enter an outer VLAN ID tag that will be imposed on the incoming frames that fulfill the match criteria. All service requests created with this setting push a dot1q outer tag on the incoming frames matching the match criteria. If a value is not provided, the push operation is ignored and not configured on the device.

**b.** In the text box **Inner VLAN ID**, enter an inner VLAN ID tag that will be imposed on the incoming frames that fulfill the match criteria. All service requests created with this setting push a dot1q inner tag on the incoming frames matching the match criteria. The Inner VLAN tag cannot be pushed without an Outer VLAN tag. That is, when pushing an Inner VLAN tag, the Outer VLAN tag also must be defined.

**Step 20** If Translate is the Rewrite Type, a **Translation Type** drop-down list appears.

The choices available in this list vary depending on the setting of the Match Inner and Outer Tags attribute (set in a previous step).

**a.** If the Match Inner and Outer Tags check box is checked (true), choose a translation type of **1:1**, **1:2**, **2:1**, or **2:2** from the Translation Type drop-down list.

- If you choose 1:1 or 2:1, enter a value in the **Outer VLAN ID** text box that appears. The outer tag of all the incoming frames that fulfill the match criteria will be translated to this ID.
- If you choose 1:2 or 2:2, enter values in the **Outer VLAN ID** and **Inner VLAN ID** text boxes that appear. The outer and inner tags of all the incoming frames that fulfill the match criteria will be translated to these IDs.

**b.** If the Match Inner and Outer Tags check box is unchecked (false), choose a translation type of **1:1** or **1:2** from the Translation Type drop-down list.

- If you choose 1:1, enter a value in the **Outer VLAN ID** text box that appears. The outer tag of all the incoming frames that fulfill the match criteria will be translated to this ID.
- If you choose 1:2, enter values in the **Outer VLAN ID** and **Inner VLAN ID** text boxes that appear. The outer and inner tags of all the incoming frames that fulfill the match criteria will be translated to these IDs.

**Step 21** Clicked **Next** to save the settings in the FlexUNI Details window.

The Standard UNI Details window appears, as shown in Figure 4-6.

**Step 22** Continue with setting the standard UNI link attributes in the next steps.

### Editing the Standard UNI Attributes

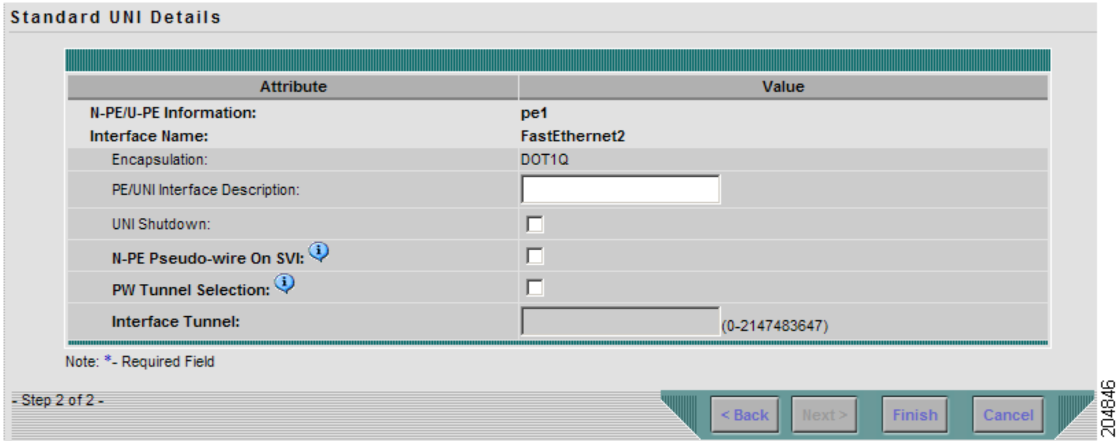
The following steps cover setting the attributes in the Standard UNI Details window. In the case of a link which is not set as a FlexUNI link (by not checking the FlexUNI check box in the Service Request Details window), editing the link attributes begins with this window.

**Note** The attributes that appear in the Standard UNI Details window are dynamically configured by ISC. Some of the attributes covered in the steps below might not appear in the window, depending on the policy and service request settings or the link type. For example, if the MPLS core connectivity type of the FlexUNI/EVC policy is VPLS or local, the pseudowire-related attributes will not appear. Also, setting the link as FlexUNI or non-FlexUNI will change the attributes that appear in the window. In addition, attributes are filtered based on device type (IOS or IOS XR). These and other cases are noted in the steps, for reference.

Some possible presentations of the Standard UNI attributes are shown in Figure 4-5 through Figure 4-7.

Figure 4-5 is an example of a direct connect link for a Cisco 7600 device running IOS with pseudowire core connectivity, configure bridge domain enabled, and the FlexUNI check box checked.

***Figure 4-5 Standard UNI Details Window (Cisco 7600, Configure Bridge Domain Enabled)***



Standard UNI Details

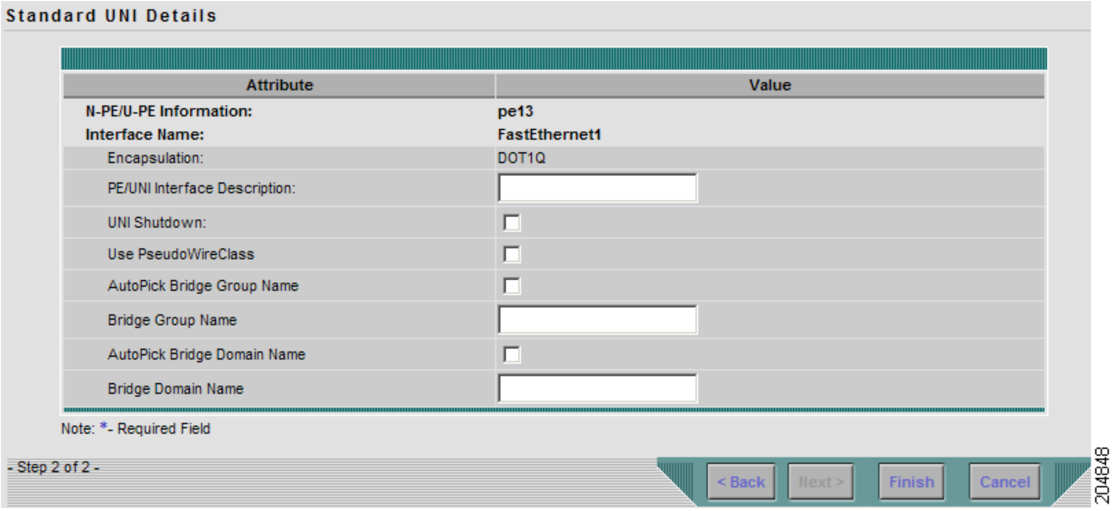
| Attribute | Value |
|---|---|
| N-PE/U-PE Information: | pe1 |
| Interface Name: | FastEthernet2 |
| Encapsulation: | DOT1Q |
| PE/UNI Interface Description: | |
| UNI Shutdown: | □ |
| N-PE Pseudo-wire On SVI: | □ |
| PW Tunnel Selection: | □ |
| Interface Tunnel: | (0-2147483647) |

Note: *- Required Field

- Step 2 of 2 -

Figure 4-6 is an example of a direct connect link for a Cisco ASR 9000 running IOS XR with pseudowire core connectivity, configure bridge domain enabled, and the FlexUNI check box checked.

***Figure 4-6 Standard UNI Details Window (Cisco ASR 9000, Configure Bridge Domain Enabled)***



Standard UNI Details

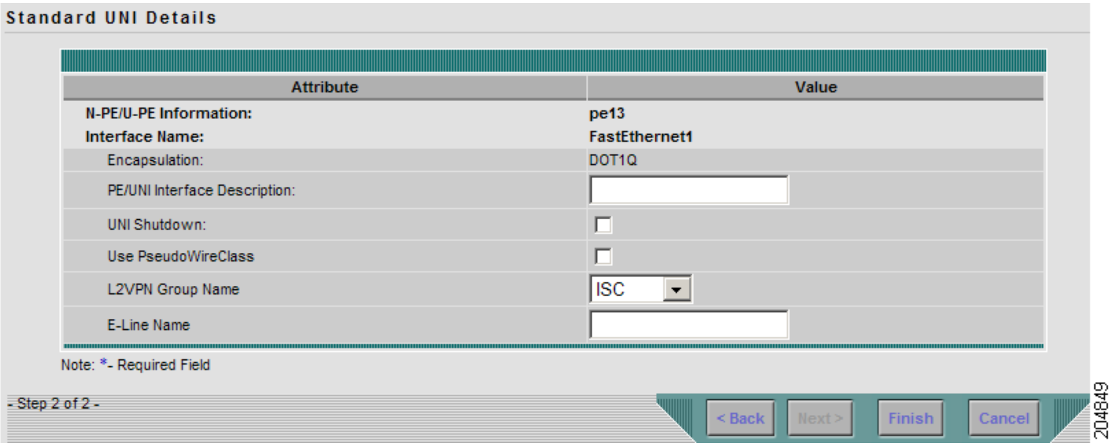
| Attribute | Value |
|---|---|
| N-PE/U-PE Information: | pe13 |
| Interface Name: | FastEthernet1 |
| Encapsulation: | DOT1Q |
| PE/UNI Interface Description: | |
| UNI Shutdown: | □ |
| Use PseudoWireClass | □ |
| AutoPick Bridge Group Name | □ |
| Bridge Group Name | |
| AutoPick Bridge Domain Name | □ |
| Bridge Domain Name | |

Note: *- Required Field

- Step 2 of 2 -

Figure 4-7 is an example of a direct connect link for a Cisco ASR 9000 running IOS XR with pseudowire core connectivity, configure bridge domain not enabled, and the FlexUNI check box checked.

*Figure 4-7 Standard UNI Details Window (Cisco ASR 9000, Configure Bridge Domain not Enabled)*



Standard UNI Details

| Attribute | Value |
|---|---|
| N-PE/U-PE Information: | pe13 |
| Interface Name: | FastEthernet1 |
| Encapsulation: | DOT1Q |
| PE/UNI Interface Description: | |
| UNI Shutdown: | □ |
| Use PseudoWireClass | □ |
| L2VPN Group Name | ISC |
| E-Line Name | |

Note: *- Required Field

- Step 2 of 2 -

**Step 23** The **N-PE/U-PE Information** and **Interface Name** fields display the PE device and interface name selected in previous steps.

These fields are read-only.

**Step 24** Choose an **Encapsulation** type from the drop-down list.

The choices are:

- **DOT1QTRUNK**—Configures the UNI as a trunk with 802.1q encapsulation. If the UNI belongs to a directly connected and FlexUNI link, this setting signifies that the incoming frames are 802.1q encapsulated and that they match the VLAN ID configured for the link. This specific topology does not involve a trunk UNI as such.
- **DOT1QTUNNEL**—Configures the UNI as an 802.1q tunnel (also known as a dot1q tunnel or Q-in-Q) port.
- **ACCESS**—Configures the UNI as an access port.

This attribute allows you to deploy different types of UNI encapsulation on different links of a service.

Usage notes:

- When a U-PE running with IOS is added in the same circuit terminating on an ASR 9000 (functioning in an N-PE role), the all three encapsulation types values will be visible in the drop-down list of the Encapsulation attribute.
- DOT1Q TUNNEL is not directly supported for ASR 9000 devices.

**Step 25** In the **PE/UNI Interface Description** field, enter a description for the interface, if desired.

**Step 26** Check the **UNI Shutdown** check box if you want to leave the UNI port shut during service activation (for example, when the service provider wants to deploy a service in the network but wants to activate it at a later time).

**Step 27** Specify the type of **VLAN Translation** for the service request by clicking the appropriate radio button.

The choices are:

- **No**—No VLAN translation is performed. (This is the default.)
- **1:1**—1:1 VLAN translation.
- **2:1**—2:1 VLAN translation.

**Note** For detailed coverage of setting up VLAN translation, see Appendix C, "Setting Up VLAN Translation."

This attribute is not displayed for IOS XR devices.

**Step 28** Check the **N-PE Pseudo-wire on SVI** check box to have ISC generate forwarding commands under SVIs (switch virtual interfaces).

By default, this check box is not checked. In this case, ISC generates forwarding commands under the service instance.

For a FlexUNI link, the attribute N-PE Pseudo-wire on SVI is dependent on the value of the attribute Configure with Bridge Domain (this is available in the service request workflow in the EVC Service Request Editor window). N-PE Pseudo-wire on SVI, if enabled, will be reflected only when Configure with Bridge Domain is set to true. Otherwise, the service request will not be created with xconnect under SVI, even if N-PE Pseudo-wire on SVI is enabled.

Usage notes:

- For a FlexUNI link, the attribute N-PE Pseudo-wire on SVI is dependent on the value of the attribute Configure with Bridge Domain (in the EVC Service Request Editor window). N-PE Pseudo-wire on SVI, if enabled, will be reflected only when Configure with Bridge Domain is set to true. Otherwise, the service request will not be created with xconnect under SVI, even if N-PE pseudo-wire on SVI is enabled.
- ISC supports a hybrid configuration for FlexUNI/EVC service requests. In a hybrid configuration, the forwarding commands (such as xconnect) for one side of an attachment circuit can be configured under a service instance, and the xconnect configuration for the other side of the attachment circuit can be configured under a switch virtual interface (SVI).
- N-PE Pseudo-wire on SVI is applicable for all connectivity types (PSEUDOWIRE, VPLS, and LOCAL), but a hybrid SVI configuration is possible only for pseudowire connectivity.
- When MPLS Core Connectivity Type is set as VPLS, the N-PE Pseudo-wire on SVI attribute is always enabled in the policy and service request.
- When MPLS Core Connectivity Type is set as LOCAL connectivity type, the N-PE Pseudo-wire on SVI attribute is always disabled in the policy and service request.
- For examples of these cases, see configlet examples FlexUNI/EVC (Pseudowire Core Connectivity, Bridge Domain, Pseudowire on SVI) and FlexUNI/EVC (Pseudowire Core Connectivity, no Bridge Domain, no Pseudowire on SVI).
- For additional information on the N-PE Pseudo-wire on SVI attribute, see the corresponding coverage in the FlexUNI/EVC policy chapter in the section Setting the Interface Attributes.
- The N-PE Pseudo-wire on SVI attribute is not supported for IOS XR devices. All the xconnect commands are configured on L2 subinterfaces.

**Step 29** Check the **PW Tunnel Selection** check box if you want to be able to manually select the Traffic Engineering (TE) tunnel for the pseudowire connecting point-to-point N-PEs.

Usage notes:

- Checking the PW Tunnel Selection check box activates the Interface Tunnel attribute field (see the next step).
- This attribute only appears if the MPLS core connectivity type is set as pseudowire in the FlexUNI/EVC policy.
- The PW Tunnel Selection attribute is not supported for IOS XR devices.

**Step 30** If you checked the PW Tunnel Selection check box, enter the TE tunnel ID in the **Interface Tunnel** text field.

Usage notes:

- ISC uses the tunnel information to create and provision a pseudowire class that describes the pseudowire connection between two N-PEs. This pseudowire class can be shared by more than one pseudowire, as long as the pseudowires share the same tunnel ID and remote loopback address. During service request creation, ISC does not check the validity of the tunnel ID number. That is, ISC does not verify the existence of the tunnel.
- The Interface Tunnel attribute is not supported for IOS XR devices.
-

**Step 31** Check the **AutoPick Bridge Group Name** check box to have ISC autopick the bridge group name during service request creation.

If this check box is unchecked, you are prompted to specify a bridge group name during service request creation (see the next step).

Usage notes:

- This attribute only displays for IOS XR devices.
- If the AutoPick Bridge Group Name check box is unchecked, enter an bridge group name in the **Bridge Group Name** text field.
- The AutoPick Bridge Group Name and Bridge Group Name attributes only appear if Configure Bridge Domain was enabled in the EVC Service Request Editor window earlier in the service request workflow.

**Step 32** Check the **AutoPick Bridge Domain/VLAN ID** check box to have ISC autopick the VLAN ID during service request creation.

If this check box is unchecked, you are prompted to specify a VLAN ID during service request creation (see the next step).

Usage notes:

- AutoPick Bridge Domain/VLAN ID consumes a global VLAN ID on the device.
- The bridge domain VLAN ID is picked from the existing ISC VLAN pool.
- The AutoPick Bridge Domain/VLAN ID attribute appears for both Cisco 7600 and ASR 9000 devices. It will be displayed only for non-FlexUNI links.

**Step 33** If the AutoPick Bridge Domain/VLAN ID check box is unchecked, enter an ID number in the **Bridge Domain/VLAN ID** text field.

Usage notes:

- If AutoPick Bridge Domain/VLAN ID is checked, this field is non-editable.
- When a VLAN ID is manually allocated, ISC verifies the VLAN ID to see if it lies within ISC's VLAN ID pool. If the VLAN ID is in the pool but not allocated, the VLAN ID is allocated to the service request. If the VLAN ID is in the pool and is already in use, ISC prompts you to allocate a different VLAN ID. If the VLAN ID lies outside of the ISC VLAN ID pool, ISC does not perform any verification about whether the VLAN ID allocated. The operator must ensure the VLAN ID is available.
- The Bridge Domain/VLAN ID text field appears for both Cisco 7600 and ASR 9000 devices. It will be displayed only for non-FlexUNI links.

**Step 34** Check the **AutoPick Bridge Domain Name** check box to have ISC autopick the bridge domain name during service request creation.

If this check box is unchecked, you are prompted to specify a bridge domain name during service request creation (see the next step).

Usage notes:

- The AutoPick Bridge Domain Name attribute appears only for Cisco ASR 9000 devices.
- The AutoPick Bridge Domain Name attribute only appears if Configure Bridge Domain was enabled in the EVC Service Request Editor window earlier in the service request workflow.

**Step 35** If the AutoPick Bridge Domain Name check box is unchecked, enter a bridge domain name in the **Bridge Domain Name** text field.

Usage notes:

- Bridge Domain Name field appears only for Cisco ASR 9000 devices.
- The Bridge Domain Name attribute only appears if Configure Bridge Domain was enabled in the EVC Service Request Editor window earlier in the service request workflow.

**Step 36** Check the **Use PseudoWireClass** check box to enable the selection of a pseudowire class.

This attribute is unchecked by default.

Usage notes:

- The pseudowire class name is used for provisioning pw-class commands on IOS XR devices. See Creating and Modifying Pseudowire Classes for IOS XR Devices, page 2-10 for additional information on pseudowire class support for IOS XR devices.
- If Use PseudoWireClass is checked, an additional attribute, **PseudoWireClass**, appears in the GUI. Click the **Select** button of PseudoWireClass attribute to choose a pseudowire class previously created in ISC.
- The Use PseudoWireClass attribute is only available if the MPLS core connectivity type was set as PSEUDOWIRE in the Service Options window (see Setting the Service Options, page 3-8).
- Use PseudoWireClass is only applicable for IOS XR devices.
- The Use PseudoWireClass and PseudoWireClass attributes only appear if Configure Bridge Domain was not enabled in the EVC Service Request Editor window earlier in the service request workflow.

**Step 37** For **L2VPN Group Name** choose one of the following from the drop-down list:

- **ISC**
- **VPNSC**

Usage notes:

- This attribute is used for provisioning the L2VPN group name on IOS XR devices.

**Note** The choices in the drop-down list are derived from a configurable DCPL property. For information about how to define the L2VPN Group Name choices available in the drop-down list, see Defining L2VPN Group Names for IOS XR Devices, page 2-14.

- The L2VPN Group Name attribute is not available if the MPLS core connectivity type was set as VPLS in the Service Options window (see Setting the Service Options, page 3-8).
- L2VPN Group Name is only applicable for IOS XR devices.
- The L2VPN Group Name attribute only appears if Configure Bridge Domain was not enabled in the EVC Service Request Editor window earlier in the service request workflow.

**Step 38** Enter an **E-Line Name** to specify the point-to-point (p2p) E-line name.

Usage notes:

- If no value is specified for the **E-Line Name**, ISC autogenerates a default name as follows:
  - For PSEUDOWIRE core connectivity type, the format is:

    *DeviceName--VC_ID*
  - For LOCAL core connectivity type, the format is:

    *DeviceName--VLAN_ID*

  If the default name is more than 32 characters, the device names are truncated.
- The E-Line Name attribute is not available if the MPLS core connectivity type was set as VPLS in the Service Options window (see Setting the Service Options, page 3-8).
- E-Line Name is only applicable for IOS XR devices.
- The E-Line Name attribute only appears if Configure Bridge Domain was not enabled in the EVC Service Request Editor window earlier in the service request workflow.

**Step 39** Click **OK** to save the Standard UNI settings and return to the EVC Service Request window.

The value in the Link Attributes column now displays as "Changed," signifying that the link settings have been updated. You can edit the link attributes now or at a future time by clicking on the Changed link and modifying the settings in the Standard UNI Details window.

See Modifying the FlexUNI/EVC Service Request, page 4-21 for details on editing the link attributes.

**Step 40** To add another link click the **Add** button and set the attributes for the new link as in the previous steps in this section.

**Step 41** To delete a link, check the check box in the first column of the row for that link and click the **Delete** button.

**Step 42** If you want to set up links with L2 access nodes for this service request, see Setting Links with L2 Access Nodes, page 4-20.

**Step 43** When you have completed setting the attributes in the EVC Service Request Editor window, click the **Save** button at the bottom of the window to save the settings and create the FlexUNI/EVC service request.

If any attributes are missing or incorrectly set, ISC displays a warning in the lower left of the window. Make any corrections or updates needed (based on the information provided by ISC), and click the **Save** button.

For information on modifying a FlexUNI/EVC service request see the section Modifying the FlexUNI/EVC Service Request, page 4-21. For additional information about saving a FlexUNI/EVC service request, see Saving the FlexUNI/EVC Service Request, page 4-23.

## Setting Links with L2 Access Nodes

The Links with L2 Access Nodes section of the EVC Service Request Editor window allows you to set up links with L2 (Ethernet) access nodes. These are similar to direct connect links, except that they have L2/Ethernet access nodes beyond the N-PE (towards the CE). Therefore, NPCs are involved. The steps for setting up links with L2 access nodes are similar to those covered in the section Setting Direct Connect Links, page 4-10. See that section for detailed steps on the following common operations:

- Adding and deleting links.
- Selecting the N-PE.

- Choosing the UNI interface.
- Setting the link as a FlexUNI link.
- Editing the standard and FlexUNI link attributes.

The main difference in setting up links with L2 access does is specifying the NPC details.

Perform the following steps to set the NPC details for links with L2 access nodes:

**Step 1** The first step in the process of adding a link using NPCs is selecting the U-PE/PE-AGG device, rather than the N-PE.

If only one NPC exists for the chosen interface, that NPC is autopopulated in the Circuit Details column, and you need not choose it explicitly.

If more then one NPC is available, click **Select one circuit** in the Circuit Selection column. The NPC window appears, enabling you to choose the appropriate NPC.

**Step 2** Click **OK**.

Each time you choose a PE and its interface, the NPC that was set up from this PE and interface is automatically displayed under **Circuit Selection**. This means that you do not have to further specify the PE to complete the link.

If you want to review the details of this NPC, click **Circuit Details** in the Circuit Details column. The NPC Details window appears and lists the circuit details for this NPC.

**Step 3** For details about editing link attributes, adding or deleting links, or using the FlexUNI check box, see the corresponding steps in the section Setting Direct Connect Links, page 4-10.

**Step 4** When you have completed setting the attributes in the EVC Service Request Editor window, click the **Save** button at the bottom of the window to save the settings and create the FlexUNI/EVC service request.

If any attributes are missing or incorrectly set, ISC displays a warning in the lower left of the window. Make any corrections or updates needed (based on the information provided by ISC), and click the **Save** button.

For information on modifying a FlexUNI/EVC service request see the section Modifying the FlexUNI/EVC Service Request, page 4-21. For additional information about saving a FlexUNI/EVC service request, see Saving the FlexUNI/EVC Service Request, page 4-23.

# Modifying the FlexUNI/EVC Service Request

You can modify a FlexUNI/EVC service request if you must change or modify the links or other settings of the service request.

To modify a FlexUNI/EVC service request, perform the following steps.

**Step 1** Choose **Service Inventory** > **Inventory and Connection Manager** > **Service Requests**.

The Service Requests window appears, showing service request available in ISC.

**Step 2** Check a check box for a service request.

**Step 3** Click **Edit**.

EVC Service Editor window appears.

**Step 4** Modify any of the attributes, as desired.

See the sections start with "Setting the Service Request Details" section on page 4-2 for detailed coverage of setting attributes in this window.

**Note** Once the VC ID, VPLS VPN ID, and VLAN ID have been set in a service request they cannot be modified.

**Step 5** To add a template/data file to an attachment circuit, see the section Using Templates and Data Files with a FlexUNI/EVC Ethernet Service Request, page 4-22.

**Step 6** When you are finished editing the FlexUNI/EVC service request, click **Save**.

For additional information about saving a FlexUNI/EVC service request, see Saving the FlexUNI/EVC Service Request, page 4-23.

# Using Templates and Data Files with a FlexUNI/EVC Ethernet Service Request

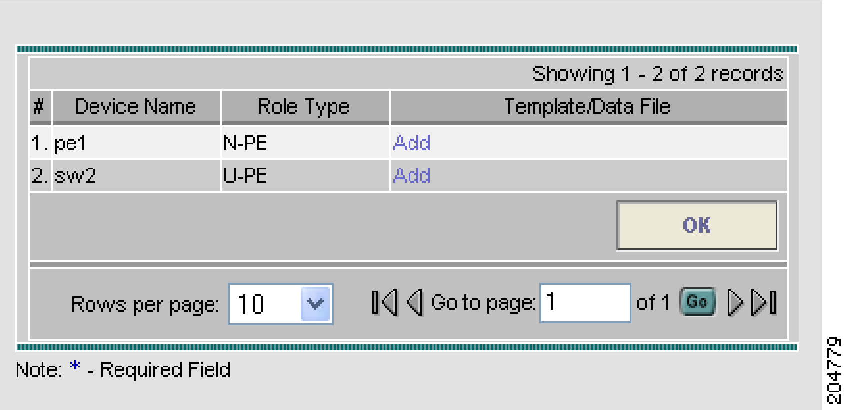
ISC does not support configuration of all the available CLI commands on a device being managed by the application. In order to configure such commands on the devices, you can use ISC Template Manager functionality. Templates can be associated at the policy level on a per-device role basis. Templates can be overridden at service request level, if the policy-level setting permits the operator to do so.

To associate templates and data files in a service request select any link in the Service Request Editor window and click the **Template** button at the bottom of the window.

**Note** If the template feature has not been enabled in the associated policy then the Template button will not be available for selection.

The SR Template Association window appears, as shown in Figure 4-8. In this window, you can associate templates at a per-device level.

*Figure 4-8 Sample Templates Association Window*



Showing 1 - 2 of 2 records

| # | Device Name | Role Type | Template/Data File |
|---|---|---|---|
| 1. | pe1 | N-PE | Add |
| 2. | sw2 | U-PE | Add |

Rows per page: 10

Go to page: 1 of 1

Note: * - Required Field

As shown in Figure 4-8, the Template Association window lists the devices comprising the link, the device roles, and the template(s)/data file(s) associated with the devices. In this case, the template(s)/data file(s) have not yet been set up.

For further instructions on how to associate templates and data files with a service request, see Appendix B, "Working with Templates and Data Files," especially the section Using Templates with Service Requests, page B-10.

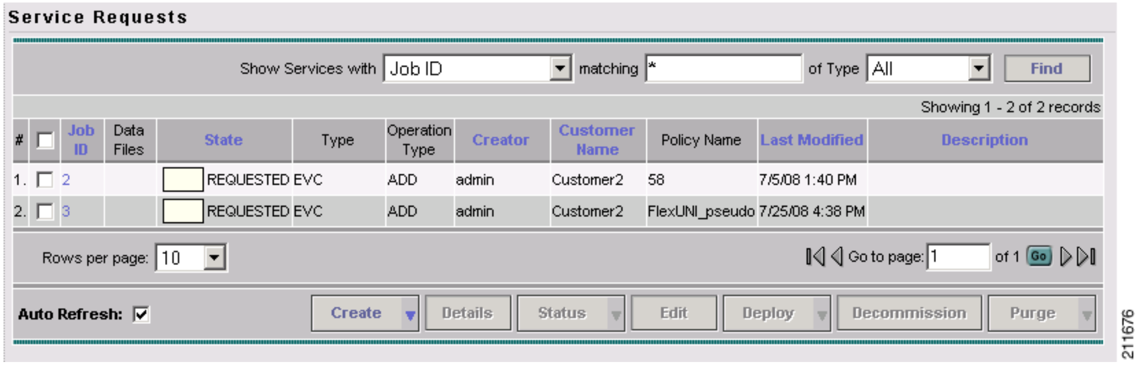
# Saving the FlexUNI/EVC Service Request

To save a FlexUNI/EVC Ethernet service request, perform the following steps.

**Step 1** When you have finished setting the attributes for the FlexUNI/EVC Ethernet service request, click **Save** to create the service request.

If the FlexUNI/EVC service request is successfully created, you will see the service request list window, similar to what appears in Figure 4-9.

*Figure 4-9 FlexUNI/EVC Service Request Created*



Service Requests

Show Services with: Job ID matching: * of Type: All

Showing 1 - 2 of 2 records

| # | □ | Job ID | Data Files | State | Type | Operation Type | Creator | Customer Name | Policy Name | Last Modified | Description |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1. | □ | 2 | | REQUESTED | EVC | ADD | admin | Customer2 | 58 | 7/5/08 1:40 PM | |
| 2. | □ | 3 | | REQUESTED | EVC | ADD | admin | Customer2 | FlexUNI_pseudo | 7/25/08 4:38 PM | |

Rows per page: 10

Go to page: 1 of 1

Auto Refresh: ☑

The newly created FlexUNI/EVC Ethernet service request is added with the state of REQUESTED, as shown in the figure.

**Step 2** If, however, the FlexUNI Ethernet service request creation failed for some reason (for example, a value chosen is out of bounds), you are warned with an error message.

In such a case, you should correct the error and save the service request again.

**Step 3** If you are ready to deploy the FlexUNI/EVC Ethernet service request, see Deploying Service Requests, page 11-1.

CHAPTER 5

# Creating a FlexUNI/EVC ATM-Ethernet Interworking Policy

This chapter contains an overview of FlexUNI/EVC ATM-Ethernet Interworking support in ISC, as well as the basic steps to create a FlexUNI/EVC ATM-Ethernet Interworking policy. It contains the following sections:

- Overview, page 5-1
- Defining the FlexUNI/EVC ATM-Ethernet Interworking Policy, page 5-2
- Setting the Service Options, page 5-4
- Setting the ATM Interface Attributes, page 5-6
- Setting the FlexUNI Attributes, page 5-7
- Setting the Interface Attributes, page 5-12
- Enabling Template Association, page 5-19

For information on creating FlexUNI/EVC ATM-Ethernet service requests, see Chapter 6, “Managing a FlexUNI/EVC ATM-Ethernet Interworking Service Request.”

**Note** For a general overview of FlexUNI/EVC support in ISC (not restricted to ATM-Ethernet Interworking), see Overview of FlexUNI/EVC Support in ISC, page 3-1.

# Overview

ISC supports interworking of a service with ATM and Ethernet protocols across the MPLS core or local switching. ATM-Ethernet interworking is supported through the following features:

- Creation of a FlexUNI/EVC policy of type “ATM-Ethernet Interworking.” The ATM-Ethernet Interworking policy type supports a choice of MPLS core options:
  - Pseudowire
  - Local (local connects)
- Provisioning of ATM-Ethernet interworking using a single FlexUNI/EVC service request.
- Combination of EVC and non-EVC syntax, that is, the creation of an L2 circuit consisting of an L2 syntax and EVC syntax.

- Supported platforms:
  - ATM interworking is supported on the Cisco 7600 with ES-20 cards.
  - ASR 9000 device is supported for IOS XR 3.7.3 and IOS XR 3.9. Because there are no ATM interfaces on the Cisco ASR 9000, ISC does not support interworking on the ASR 9000 for ATM interfaces. Only Ethernet interfaces are supported.

# Defining the FlexUNI/EVC ATM-Ethernet Interworking Policy

You must define a FlexUNI/EVC ATM-Ethernet Interworking policy before you can provision a service. A policy can be shared by one or more service requests that have similar service requirements.

A policy is a template of most of the parameters needed to define a FlexUNI/EVC service request. After you define it, a FlexUNI/EVC policy can be used by all the FlexUNI/EVC service requests that share a common set of characteristics. You create a new FlexUNI/EVC policy whenever you create a new type of service or a service with different parameters. FlexUNI/EVC policy creation is normally performed by experienced network engineers.

The Editable check box for an attribute in the policy gives the network operator the option of making a field editable. If the value is set to editable, the service request creator can change the value(s) of the particular policy attribute. If the value is *not* set to editable, the service request creator cannot change the attribute.

You can also associate Cisco IP Solution Center (ISC) templates and data files with a service request. See Appendix B, "Working with Templates and Data Files," for more about using templates and data files in service requests.

To define a FlexUNI/EVC ATM-Ethernet Interworking policy, you start by setting the service type attributes. To do this, perform the following steps.

---

**Step 1** Choose **Service Design** > **Policies**.

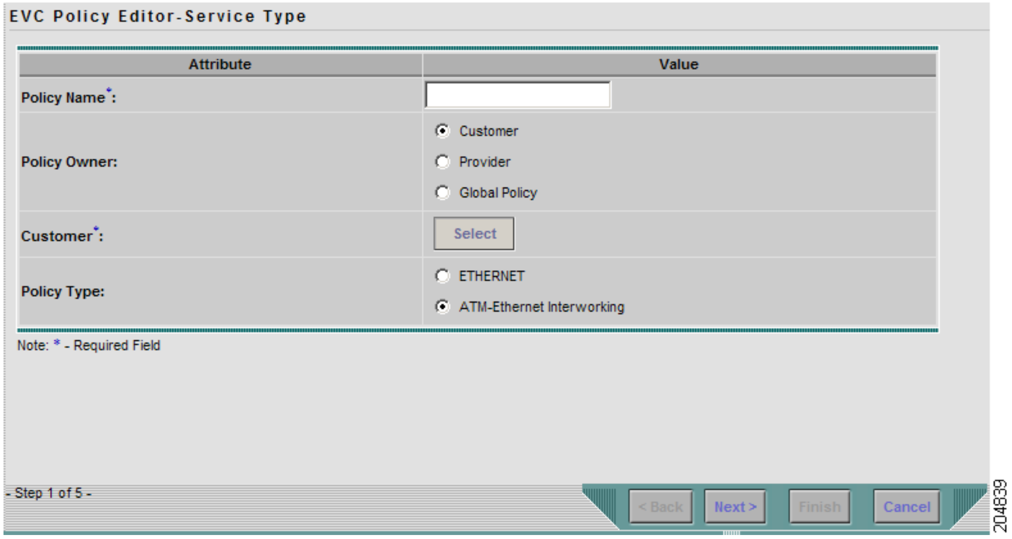
The Policies window appears.

**Step 2** Click **Create**.

**Step 3** Choose **FlexUNI (EVC) Policy**.

The EVC Policy Editor - Service Type window appears, as shown in Figure 5-1.

*Figure 5-1 EVC Policy Editor - Service Type*



EVC Policy Editor-Service Type

| Attribute | Value |
|---|---|
| Policy Name*: | |
| Policy Owner: | ● Customer<br>○ Provider<br>○ Global Policy |
| Customer*: | Select |
| Policy Type: | ○ ETHERNET<br>● ATM-Ethernet Interworking |

Note: * - Required Field

- Step 1 of 5 -

**Step 4** Enter a **Policy Name** for the FlexUNI/EVC ATM-Ethernet Interworking policy.

**Step 5** Choose the **Policy Owner** for the FlexUNI/EVC policy.

There are three types of FlexUNI/EVC policy ownership:

- Customer ownership
- Provider ownership
- Global ownership—Any service operator can make use of this policy.

This ownership has relevance when the ISC Role-Based Access Control (RBAC) comes into play. For example, a FlexUNI/EVC policy that is customer-owned can only be seen by operators who are allowed to work on this customer-owned policy. Similarly, operators who are allowed to work on a provider's network can view, use, and deploy a particular provider-owned policy.

**Step 6** Click **Select** to choose the owner of the FlexUNI/EVC policy.

The policy owner was established when you created customers or providers during ISC setup. If the ownership is global, the Select function does not appear.

**Step 7** Choose the **Policy Type**.

The choices are:

- **ETHERNET**
- **ATM-Ethernet Interworking**

**Note** This chapter describes creating the ATM-Ethernet Interworking policy type. For information on using the FlexUNI/EVC ETHERNET policy type, see Chapter 3, "Creating a FlexUNI/EVC Ethernet Policy."

**Step 8** Click **Next**.

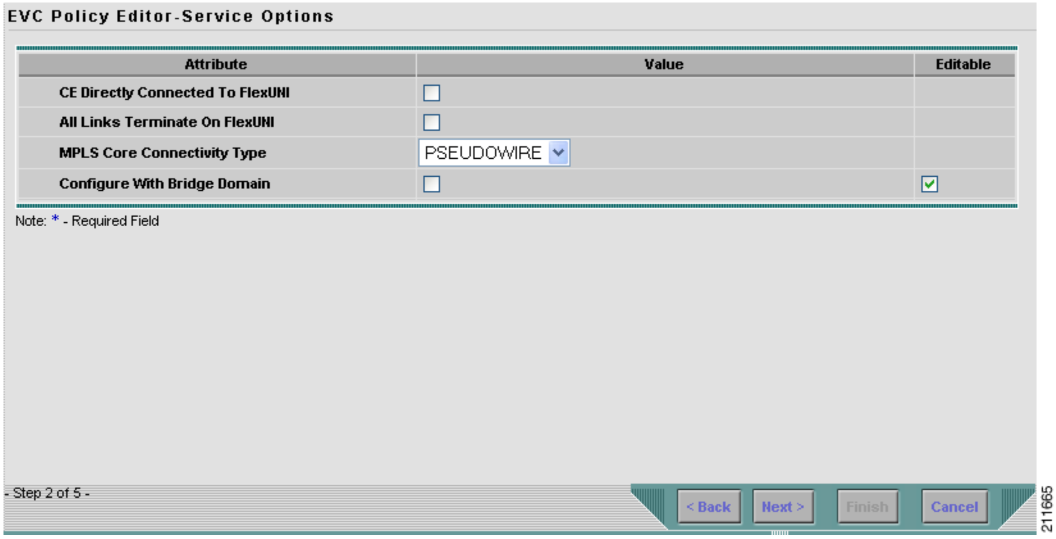
The EVC Policy Editor - Service Options window appears, as show in Figure 5-2.

**Step 9** Continue with the steps contained in the next section, Setting the Service Options, page 5-4.

# Setting the Service Options

This section describes how to set the service options for the FlexUNI/EVC policy, as shown in Figure 5-2.

*Figure 5-2 EVC Policy Editor - Service Options Window*



EVC Policy Editor-Service Options

| Attribute | Value | Editable |
|---|---|---|
| CE Directly Connected To FlexUNI | ☐ | |
| All Links Terminate On FlexUNI | ☐ | |
| MPLS Core Connectivity Type | PSEUDOWIRE | |
| Configure With Bridge Domain | ☐ | ☑ |

Note: * - Required Field

- Step 2 of 5 -

The **Editable** check box gives you the option of making a field editable. If you check the **Editable** check box, the service operator who is using this FlexUNI/EVC policy can modify the editable parameter during FlexUNI/EVC service request creation.

To set the FlexUNI/EVC service options, perform the following steps.

**Step 1** Check the **CE Directly Connected to FlexUNI** check box if the CEs are directly connected to the N-PE.

This check box is not checked by default.

Usage notes:

- If the check box is checked, a service request created using this policy can have only directly connected links. No Ethernet access nodes will be involved.
- If the check box is unchecked, a service request created using this policy might or might not have Ethernet access nodes in the links.
- When a CE is directly connected to the N-PE, NPCs are not applicable to the link while creating service requests.
- When a CE is not directly connected to the N-PE, NPCs are used during service request creation, as per standard ISC behavior. There is no change in NPC implementation to support FlexUNI/EVC functionality.

**Step 2** Check the **All Links Terminate on FlexUNI** check box if all links need to be configured with FlexUNI/EVC features.

This check box is not check by default. Usage notes:

- If the check box is checked, a service request created using such policy will have all links using the FlexUNI/EVC feature.

- If the check box is unchecked, zero or more links can use the FlexUNI/EVC feature. This ensures that existing platforms can still be used in one or more links while delivering the services. This allows the possibility of a link with FlexUNI/EVC support being added in the future.

  **Note** If the check box is unchecked, in the service request creation process the user must indicate whether or not the created link is FlexUNI or non-FlexUNI.

- If no links are expected to use the FlexUNI/EVC feature even in the future (for example, if the provider is not planning to upgrade to the EVC infrastructure for the service that is being created), existing ISC policy types (L2VPN or VPLS) can be used instead of FlexUNI/EVC.

**Step 3** Choose an **MPLS Core Connectivity Type** from the drop-down list.

**Note** The core option supports MPLS only. There is no L2TPv3 support for this service.

The choices are:

- **PSEUDOWIRE**—Choose this option to allow connectivity between two N-PEs across the MPLS core. This option does not limit the service to point-to-point (E-Line). This is because even with the PSEUDOWIRE option selected, there can still be multiple CEs connected to a bridge domain on one or both sides of the pseudowire.
- **LOCAL**—Choose this option for local connect cases in which there is no connectivity required across the MPLS core.

  Local connect supports the following scenarios:

  – All interfaces on the N-PE are FlexUNI-capable and using the EVC infrastructure. This is configured by associating all of the customer traffic on these interfaces to a bridge domain. This consumes a VLAN ID on the N-PE (equal to the bridge domain ID).

  – Some interfaces on the N-PE are FlexUNI-capable, while others are switch-port-based. In such cases, all of the customer traffic on the interfaces that are configured with the EVC infrastructure are associated to a bridge domain. The traffic on the non-FlexUNI interfaces (and all the access nodes/interfaces beyond this N-PE) are configured with the Service Provider VLAN ID, where the Service Provider VLAN ID is the same as the bridge domain ID for the EVC-based services.

  – Only two interfaces on the N-PE are involved, and both are based on FlexUNI-capable line cards. In the first case, the operator might choose not to configure the bridge domain option. In this case, the **connect** command that is used for the local connects are used, and the global VLAN is conserved on the device. If the operator chooses to configure with the bridge domain option, both interfaces are associated to a bridge domain ID, so that additional local links can be added to the service in future. This consumes a VLAN ID (bridge domain ID) on the N-PE.
- **VPLS**—This option is not supported for FlexUNI/EVC ATM-Ethernet Interworking policies and services requests.

**Note** Attributes available in subsequent windows of the policy workflow dynamically change based on the choice made for the MPLS Core Connectivity Type (PSEUDOWIRE or LOCAL). For completeness, all attributes available for the different core types are documented in the following steps. Attributes apply to all core types, unless otherwise noted.

**Note** Also, some attributes are supported only on IOS or IOS XR platforms. Attributes apply to both platforms, unless otherwise noted. All platform-specific attributes are visible in the policy workflow windows. Later, when a service request is created based on the policy (and specific devices are associated with the service request), platform-specific attributes are filtered from service request windows, depending on the device type (IOS or IOS XR).

**Step 4** Check the **Configure With Bridge Domain** check box to determine bridge domain characteristics.

The behavior of the Configure With Bridge-Domain option works in tandem with the choice you selected in the MPLS Core Connectivity Type option, as follows.

- **PSEUDOWIRE** as the MPLS Core Connectivity Type. There are two cases:

  A. With FlexUNI:

  – If **Configure With Bridge Domain** is checked, the policy configures pseudowires under SVIs associated to the bridge domain.

  – If **Configure With Bridge Domain** is unchecked, the policy will configure pseudowires directly under the service instance. This conserves the global VLAN.

  B. Without FlexUNI:

  – If **Configure With Bridge Domain** is checked, the policy configures pseudowires as in L2VPN services (with SVIs).

  – If **Configure With Bridge Domain** is unchecked, the policy configures pseudowires directly under subinterfaces.

  Only pseudowires can be either configured directly under service instance of the corresponding FlexUNI-capable interface or under SVIs associated to the bridge domain.

- **LOCAL** as the MPLS Core Connectivity Type:

  – If **Configure With Bridge Domain** is checked, the policy allows either point-to-point or multipoint local connect services.

  – If **Configure With Bridge Domain** is unchecked, ISC allows only point-to-point local connects without bridge domain.

**Step 5** Click **Next**.

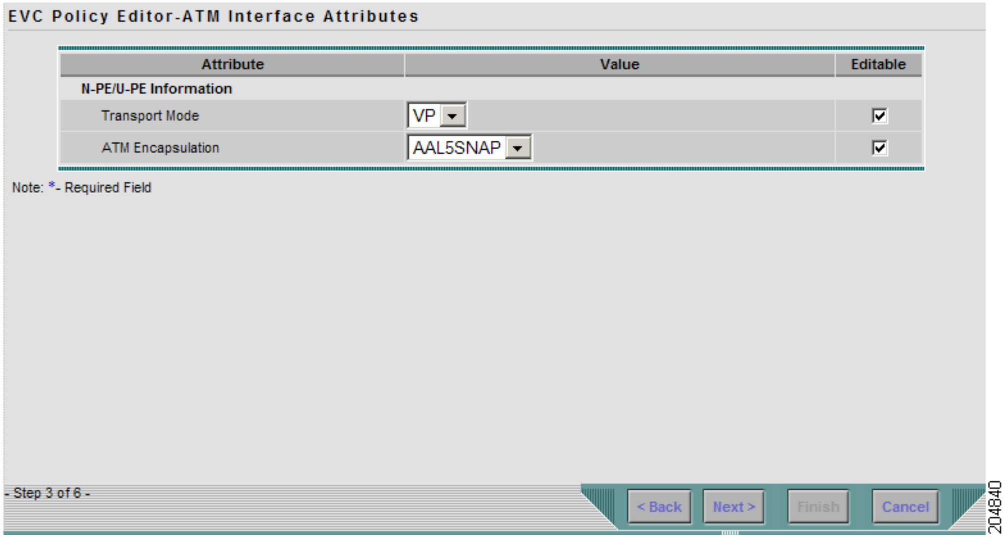
The EVC Policy Editor - ATM Interface Attribute window appears, as shown in Figure 5-3.

**Step 6** Continue with the steps contained in the next section, Setting the ATM Interface Attributes, page 5-6.

# Setting the ATM Interface Attributes

This section describes how to set the ATM Interface attributes for the FlexUNI/EVC ATM-Ethernet Interworking policy, as shown in Figure 5-3.

*Figure 5-3 EVC Policy Editor - ATM Interface Attributes Window*



EVC Policy Editor-ATM Interface Attributes

| Attribute | Value | Editable |
|---|---|---|
| N-PE/U-PE Information | | |
| Transport Mode | VP | ☑ |
| ATM Encapsulation | AAL5SNAP | ☑ |

Note: *- Required Field

- Step 3 of 6 -

To set the ATM interface attributes, perform the following steps.

**Step 7** Choose the **Transport Mode** from the drop-down list.

The choices are:

- **VP**—Virtual path mode. This is the default.
- **VC**—Virtual circuit mode.

**Step 8** Choose the **ATM Encapsulation** from the drop-down list.

- **AAL5SNAP**

**Step 9** Click **Next**.

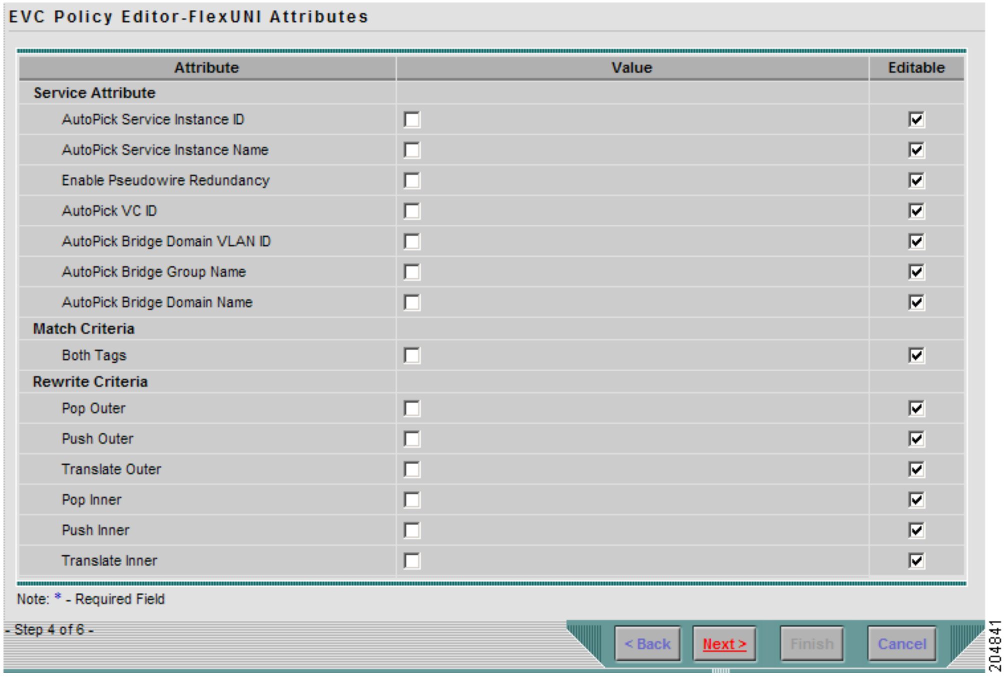
The EVC Policy Editor - FlexUNI Attribute window appears, as shown in Figure 5-4.

**Step 10** Continue with the steps contained in the next section, Setting the FlexUNI Attributes, page 5-7.

# Setting the FlexUNI Attributes

This section describes how to set the FlexUNI attributes for the FlexUNI/EVC ATM-Ethernet Interworking policy, as shown in Figure 5-4.

*Figure 5-4 EVC Policy Editor - FlexUNI Attribute Window*



EVC Policy Editor-FlexUNI Attributes

| Attribute | Value | Editable |
|---|---|---|
| **Service Attribute** | | |
| AutoPick Service Instance ID | □ | ☑ |
| AutoPick Service Instance Name | □ | ☑ |
| Enable Pseudowire Redundancy | □ | ☑ |
| AutoPick VC ID | □ | ☑ |
| AutoPick Bridge Domain VLAN ID | □ | ☑ |
| AutoPick Bridge Group Name | □ | ☑ |
| AutoPick Bridge Domain Name | □ | ☑ |
| **Match Criteria** | | |
| Both Tags | □ | ☑ |
| **Rewrite Criteria** | | |
| Pop Outer | □ | ☑ |
| Push Outer | □ | ☑ |
| Translate Outer | □ | ☑ |
| Pop Inner | □ | ☑ |
| Push Inner | □ | ☑ |
| Translate Inner | □ | ☑ |

Note: * - Required Field

- Step 4 of 6 -

FlexUNI attributes are organized under the following categories:

- Service Attributes
- VLAN Match Criteria
- VLAN Rewrite Criteria

The following sections describe how to set the options under each category.

# Setting the Service Attributes

To set the FlexUNI service attributes, perform the following steps.

**Step 1** Check the **AutoPick Service Instance ID** check box to specify that the service instance ID will be autogenerated and allocated to the link during service request creation.

If the check box is unchecked, while setting the ISC link attributes during service request creation, ISC will prompt the operator to specify the service instance ID.

Usage notes:

- The service instance ID represents an Ethernet Flow Point (EFP) on an interface in the EVC infrastructure. The service instance ID is locally significant to the interface. This ID has to be unique only at the interface level. The ID must be a value from 1 to 8000.
- There are no resource pools available in ISC from which to allocate the service instance IDs.
- It is the responsibility of the operator creating the service request to maintain the uniqueness of the ID at the interface level.

**Step 2** Check the **AutoPick Service Instance Name** check box to have ISC autogenerate a service instance name when you create a service request based on the policy. The autogenerated value is in the following pattern: *CustomerName_ServiceRequestJobID*.

If the check box is unchecked, then you can enter a value during service request creation.

**Step 3** Check the **Enable PseudoWire Redundancy** check box to enable pseudowire redundancy (alternative termination device) under certain conditions.

Usage notes:

- Enable Pseudo Wire Redundancy is only available if the MPLS Core Connectivity Type was set as PSEUDOWIRE in the Service Options window (see Setting the Service Options, page 5-4).

**Step 4** Check the **AutoPick VC ID** check box to have ISC autopick the VC ID during service request creation.

If this check box is unchecked, the operator will be prompted to specify a VC ID during service request creation.

Usage notes:

- When AutoPick VC ID is checked, ISC allocates a VC ID for pseudowires from the ISC-managed VC ID resource pool.

**Step 5** Check the **AutoPick Bridge Domain/VLAN ID** check box to have ISC autopick the VLAN ID for the service request during service request creation.

If this check box is unchecked, the operator will be prompted to specify a VLAN ID during service request creation.

Usage notes:

- AutoPick Bridge Domain/VLAN ID consumes a global VLAN ID on the device.
- The bridge domain/VLAN ID is picked from the existing ISC VLAN pool. Once the VLAN ID is assigned in the service request, ISC makes the VLAN ID unavailable for subsequent service requests.
- In the case of manual VLAN ID allocation, ISC does not manage the VLAN ID if the ID lies outside the range of an ISC-managed VLAN pool. In this case, the operator must ensure the uniqueness of the ID in the Ethernet access domain. If an operator specifies a VLAN ID that is within the range of an ISC-managed VLAN pool and the VLAN ID is already in use in the access domain, ISC displays an error message indicating that the VLAN ID is in use.

### Note on Access VLAN IDs

An access VLAN ID is of local significance to the FlexUNI-capable ports. It should not be confused with the global VLANs. This can be visualized as a partitioning of the Ethernet access network beyond the FlexUNI ports into several subEthernet access domains (one each for a FlexUNI-capable port).

However, all the service interfaces on the Ethernet access nodes beyond the FlexUNI ports will have this very same VLAN ID for a link. This ID must be manually specified by the operator when setting the link attributes during service request creation. The operator must ensure the uniqueness of the ID across the FlexUNI-demarcated Ethernet access domain.

These VLAN IDs are not managed by ISC by means of locally-significant VLAN pools. But once a VLAN ID is assigned for a link in the service request, ISC makes the VLAN unavailable for subsequent service requests within the Ethernet access domain demarcated by the FlexUNI. Likewise, if a manually-specified VLAN is already in use in the access domain delimited by the FlexUNI, ISC will display an error message indicating that the new VLAN ID being specified is already in use on the NPC. The operator will be prompted to specify a different VLAN ID, which will be provisioned on the L2 access nodes.

**Step 6** Continue with the steps contained in the next section, Setting the VLAN Matching Criteria Attributes, page 5-10.

## Setting the VLAN Matching Criteria Attributes

Prior to the introduction of the FlexUNI capability, service providers could either deploy service-multiplexed services (ERS/ERMS or EVPL/EVCS) or service-bundled services on a single port. Both could not be supported simultaneously due to the limitations in the infrastructure, which only allowed matching the outer-most VLAN tag.

One of the key benefits of FlexUNI/EVC support in ISC is to provide a flexible means to examine the VLAN tags (up to two levels) of the incoming frames and associate them to appropriate Ethernet Flow Points (EFPs). This allows service providers to deploy simultaneously both the service-multiplexed and service-bundled services on a single port.

To set the FlexUNI VLAN matching criteria attributes, perform the following steps.

**Step 1** Check the **Both Tags** check box to enable service requests created with the policy to match both the inner and outer VLAN tags of the incoming frames.

If you do not check this check box, service requests created with the policy will match only the outer VLAN tag of the incoming frames.

Checking the Both Tags attribute causes the Inner VLAN Ranges attribute (covered in the next steps) to appear in the FlexUNI Attribute window.

**Step 2** Check the **Inner VLAN Ranges** check box to enable the range of inner VLAN tags to be specified during service request creation.

If the check box is unchecked, the range of inner VLAN tags are not allowed. In this case, the operator must specify discrete VLAN IDs during service request creation.

**Step 3** Continue with the steps contained in the next section, Setting the VLAN Rewrite Criteria Attributes, page 5-10.

## Setting the VLAN Rewrite Criteria Attributes

Together with VLAN matching criteria, VLAN rewrite makes the FlexUNI/EVC infrastructure very powerful and flexible. The following VLAN rewrite options are supported:

- Pop one or two tags.
- Push one or two tags.
- Translation (1:1, 2:1, 1:2, 2:2).

Be aware of the following considerations when setting the VLAN rewrite criteria attributes:

- Only one kind of rewrite can be done on every CE-facing FlexUNI link.
- All VLAN rewrites are done using the **symmetric** keyword on the ingress traffic (for example, **rewrite ingress tag pop 2 symmetric**).

- For any service instance, only one type of rewrite option (pop, push, or translate) is allowed per instance. For example, if pop out is enabled, push inner, push outer, translate inner, and translate outer are not available.

To set the FlexUNI VLAN rewrite criteria attributes, perform the following steps.

**Step 1** Check the **Pop Outer** check box to pop the outer VLAN ID tag of the incoming frames that fulfill the match criteria.

If this check box is unchecked, the outer tag of the incoming traffic is not popped.

**Step 2** Check the **Pop Inner** check box to pop the inner VLAN ID tag of the incoming frames that fulfill the match-criteria.

If this check box is unchecked, the inner tag is not popped. Note that, if Pop Inner is checked, Pop Outer is automatically checked.

**Step 3** Check the **Push Outer** check box to impose an outer VLAN ID tag onto the incoming frames that fulfill the match criteria.

If this check box is unchecked, no outer tag is imposed on the incoming frames.

Usage notes:

- If Push Outer is checked, all service requests created with the policy push a dot1q outer tag on the incoming frames matching the match criteria. When creating the link during service creation, the operator can specify an outer tag with a value from 1 to 4096.
- This attribute is available regardless of the number of tags used in the match criteria. Whether the incoming traffic is double tagged or single tagged, if Push Outer is enabled, all corresponding service requests push an outer tag. All subsequent nodes consider only the outer-most two tags (if FlexUNI-capable) or just one tag (not FlexUNI-capable) and treat the inner-most tags transparently as payload.
- This VLAN ID is not derived from ISC-managed VLAN ID pools.

**Step 4** Check the **Push Inner** check box to impose an inner VLAN ID tag onto the incoming frames that fulfill the match criteria.

This operation pushes both an inner and an outer tag onto the incoming packet, not just an inner tag. If this check box is unchecked, no inner tag is imposed on the incoming frames.

Usage notes:

- If Push Inner is checked, all service requests created with the policy push a dot1q inner tag on the incoming frames matching the match criteria. When creating the link during service creation, the operator can specify an inner tag with a value from 1 to 4096.
- If Push Inner is checked, Push Outer is automatically checked.
- This attribute is available regardless of the number of tags used in the match criteria. Regardless of whether the incoming traffic is double tagged or single tagged, if Push Inner is enabled, all corresponding service requests push an inner tag. All subsequent nodes consider only the outer-most two tags (if FlexUNI-capable) or just one tag (not FlexUNI-capable) and treat the inner-most tags transparently as payload.
- This VLAN ID is not derived from ISC-managed VLAN ID pools.

**Step 5** Check the **Translate Outer** check box to allow the operator to specify a target outer VLAN ID during service request creation.

The outer tag of all the incoming frames that fulfill the match criteria are translated to this ID. If the check box is unchecked, no outer tag translation is performed. See Table 5-1.

**Step 6** Check the **Translate Inner** check box to allow the operator to specify a target inner VLAN ID during service request creation.

The inner tag of all the incoming frames that fulfill the match criteria are translated to this ID. If the check box is unchecked, no inner tag translation is performed. See Table 5-1.

**Note** Table 5-1 summarizes the realization of different VLAN translations available in the FlexUNI/EVC infrastructure. The second and third columns (Match Outer Tag and Match Inner Tag) refer to policy settings. The last two columns (Translate Outer Tag and Translate Inner Tag) indicate the VLAN translation that occurs on the incoming frames.

*Table 5-1 VLAN Translation Summary Table*

| Type | Match Outer Tag | Match Inner Tag | Translate Outer Tag | Translate Inner Tag |
|---|---|---|---|---|
| 1:1 | True | N/A | Yes | No |
| 1:2 | True | N/A | Yes | Yes |
| 2:1 | True | True | Yes | No |
| 2:2 | True | True | Yes | Yes |

**Step 7** Click **Next**.

The EVC Policy Editor - Interface Attribute window appears, as shown in Figure 5-4.

**Step 8** Continue with the steps contained in the next section, Setting the Interface Attributes, page 5-12.

# Setting the Interface Attributes

This step of creating the FlexUNI/EVC ATM-Ethernet Interworking policy involves setting the interface attributes, as shown in the EVC Policy Editor - Interface Attribute window in Figure 5-5. The attributes you can configure in this window are grouped under the following categories:

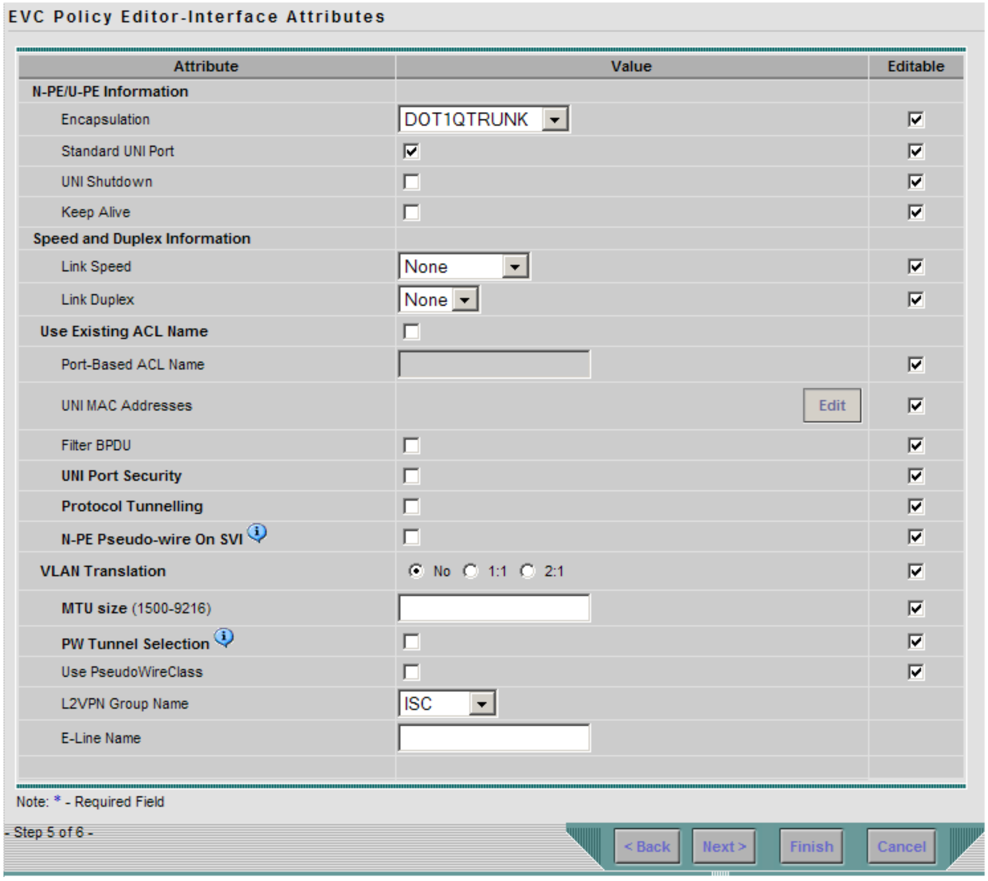
- N-PE/U-PE information
- Speed and duplex information
- ACL name and MAC addresses
- UNI port security
- Storm control
- L2 protocol tunneling

In some cases, checking an attribute causes additional attributes to appear in the GUI. This is covered in the steps that follow.

**Note** If the CE is directly connected to an N-PE, only speed, duplex, UNI shutdown, and other generic options are presented. In this case, port security, storm control, L2 protocol tunneling, and other advanced features are not supported due to the current platform limitations. If these features are needed for a service, the service provider must deploy Layer 2 Ethernet access nodes beyond the FlexUNI to support these requirements.

**Note** Attributes available in the Interface Attributes window dynamically change based on the choice made for the MPLS Core Connectivity Type (PSEUDOWIRE or LOCAL) in the Service Options window (see Setting the Service Options, page 5-4). For completeness, all attributes available for the different core types are documented in the following steps. Attributes apply to all core types, unless otherwise noted.

*Figure 5-5 EVC Policy Editor - Interface Attributes Window*



EVC Policy Editor-Interface Attributes

| Attribute | Value | Editable |
|---|---|---|
| N-PE/U-PE Information | | |
| Encapsulation | DOT1QTRUNK ▾ | ☑ |
| Standard UNI Port | ☑ | ☑ |
| UNI Shutdown | □ | ☑ |
| Keep Alive | □ | ☑ |
| Speed and Duplex Information | | |
| Link Speed | None ▾ | ☑ |
| Link Duplex | None ▾ | ☑ |
| Use Existing ACL Name | □ | |
| Port-Based ACL Name | | ☑ |
| UNI MAC Addresses | Edit | ☑ |
| Filter BPDU | □ | ☑ |
| UNI Port Security | □ | ☑ |
| Protocol Tunnelling | □ | ☑ |
| N-PE Pseudo-wire On SVI ⓘ | □ | ☑ |
| VLAN Translation | ● No ○ 1:1 ○ 2:1 | ☑ |
| MTU size (1500-9216) | | ☑ |
| PW Tunnel Selection ⓘ | □ | ☑ |
| Use PseudoWireClass | □ | ☑ |
| L2VPN Group Name | ISC ▾ | |
| E-Line Name | | |

Note: * - Required Field

- Step 5 of 6 -

To set the FlexUNI/EVC interface attributes, perform the following steps.

**Step 1** Choose an **Encapsulation** type.

The choices are:

- **DOT1QTRUNK**—Configures the UNI as a trunk with 802.1q encapsulation. If the UNI belongs to a directly connected and FlexUNI link, this setting signifies that the incoming frames are 802.1q encapsulated and that they match the VLAN ID configured for the link. This specific topology does not involve a trunk UNI as such.
- **DOT1QTUNNEL**—Configures the UNI as an 802.1q tunnel (also known as a dot1q tunnel or Q-in-Q) port.
- **ACCESS**—Configures the UNI as an access port.

**Step 2** Check the **Standard UNI Port** check box to enable port security.

This is the default. When you uncheck the check box, the port is treated as an uplink with no security features, and the window dynamically changes to eliminate items related to port security.

**Note** When the UNI is configured on an N-PE device running IOS XR, the Standard UNI Port attribute is not supported. All the CLIs related to Standard UNI Port and UNI Port Security are ignored in this case.

**Step 3** Check the **UNI Shutdown** check box if you want to leave the UNI port shut during service activation, for example, when the service provider wants to deploy a service in the network but wants to activate it at a later time.

**Step 4** Check the **Keep Alive** check box to configure keepalives on the UNI port.

By default, this check box is unchecked, which causes the command **no keepalive** to be provisioned on the UNI port. This prevents a CPE from sending keepalive packets to the U-PE, for security purposes. This attribute is editable, in order to support modification on a per-service request basis.

**Step 5** Enter a **Link Speed** (optional) of None, 10, 100, 1000, Auto, or nonegotiate.

**Step 6** Enter a **Link Duplex** (optional) of None, Full, Half, or Auto.

**Step 7** Check the **Use Existing ACL Name** check box if you want to assign your own named access list to the port.

By default, this check box is not checked and ISC automatically assigns a MAC-based ACL on the customer facing UNI port, based on values you enter in **UNI MAC addresses** (below).

**Step 8** Enter a **Port-Based ACL Name** (if you checked the **Use Existing ACL Name** check box, as mentioned in the previous step).

**Note** ISC does not create this ACL automatically. The ACL must already exist on the device, or be added as part of a template, before the service request is deployed. Otherwise, deployment will fail.

**Step 9** Enter one or more Ethernet MAC addresses in **UNI MAC addresses**.

This selection is present only if you uncheck the **Use Existing ACL Name** check box. Click the **Edit** button to bring up a pop-up window in which you enter MAC addresses to be allowed or denied on the port. You can also specify a range of addresses by setting a base MAC address and a filtered MAC address.

**Step 10** Check the **UNI Port Security** check box (see Figure 5-6) if you to want to provision port security-related CLIs to the UNI port by controlling the MAC addresses that are allowed to go through the interface.

a. For **Maximum Number of MAC address**, enter the number of MAC addresses allowed for port security.

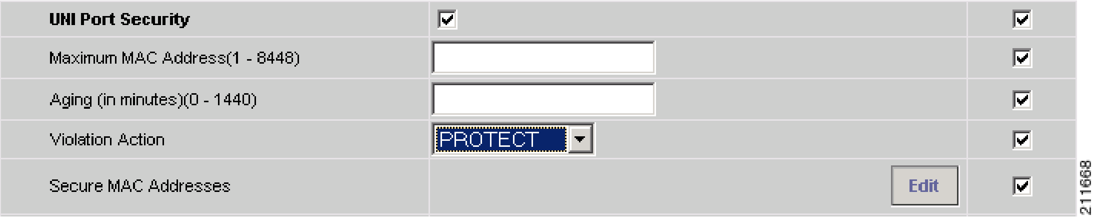
b. For **Aging,** enter the length of time the MAC address can stay on the port security table.

c. For **Violation Action**, choose what action will occur when a port security violation is detected:

- **PROTECT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value.
- **RESTRICT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value and causes the Security Violation counter to increment.
- **SHUTDOWN**—Puts the interface into the error-disabled state immediately and sends an SNMP trap notification.

**d.** In the **Secure MAC Addresses** field, enter one or more Ethernet MAC addresses.

*Figure 5-6 UNI Port Security*

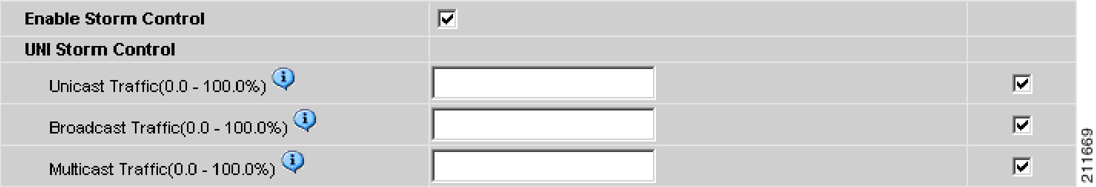


| | | |
|---|---|---|
| **UNI Port Security** | ☑ | ☑ |
| Maximum MAC Address(1 - 8448) | | ☑ |
| Aging (in minutes)(0 - 1440) | | ☑ |
| Violation Action | PROTECT ▾ | ☑ |
| Secure MAC Addresses | Edit | ☑ |

**Step 11** Check the **Enable Storm Control** check box (see Figure 5-7) to help prevent the UNI port from being disrupted by a broadcast, multicast, or unicast storm.

Enter a threshold value for each type of traffic. The value, which can be specified to two significant digits, represents the percentage of the total available bandwidth of the port. If the threshold of a traffic type is reached, further traffic of that type is suppressed until the incoming traffic falls below the threshold level.

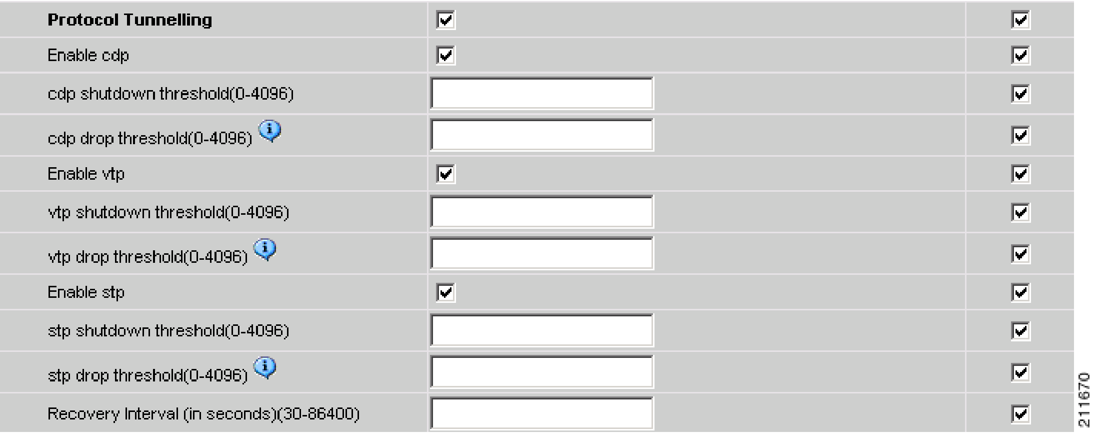
*Figure 5-7 Enable Storm Control*



| | | |
|---|---|---|
| **Enable Storm Control** | ☑ | |
| **UNI Storm Control** | | |
| Unicast Traffic(0.0 - 100.0%) ⓘ | | ☑ |
| Broadcast Traffic(0.0 - 100.0%) ⓘ | | ☑ |
| Multicast Traffic(0.0 - 100.0%) ⓘ | | ☑ |

**Step 12** Check the **Protocol Tunnelling** check box (see Figure 5-8) if you want to define the Layer 2 Bridge Protocol Data Unit (BPDU) frames that can be tunneled over the core to the other end.

*Figure 5-8 Protocol Tunnelling*



| | | |
|---|---|---|
| **Protocol Tunnelling** | ☑ | ☑ |
| Enable cdp | ☑ | ☑ |
| cdp shutdown threshold(0-4096) | | ☑ |
| cdp drop threshold(0-4096) ⓘ | | ☑ |
| Enable vtp | ☑ | ☑ |
| vtp shutdown threshold(0-4096) | | ☑ |
| vtp drop threshold(0-4096) ⓘ | | ☑ |
| Enable stp | ☑ | ☑ |
| stp shutdown threshold(0-4096) | | ☑ |
| stp drop threshold(0-4096) ⓘ | | ☑ |
| Recovery Interval (in seconds)(30-86400) | | ☑ |

For each protocol that you choose, enter the shutdown threshold and drop threshold for that protocol:

**a.** **Enable cdp**—Enable Layer 2 tunnelling on Cisco Discover Protocol (CDP).

b. **cdp shutdown threshold—**Enter the number of packets per second to be received before the interface is shut down.

c. **cdp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping CDP packets.

d. **Enable vtp**—Enable Layer 2 tunnelling on VLAN Trunk Protocol (VTP).

e. **vtp shutdown threshold**—Enter the number of packets per second to be received before the interface is shut down.

f. **vtp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping VTP packets.

g. **Enable stp—**Enable Layer 2 tunnelling on Spanning Tree Protocol (STP).

h. **stp shutdown threshold—**Enter the number of packets per second to be received before the interface is shut down.

i. **stp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping STP packets.

j. **Recovery Interval**—Enter the amount of time, in seconds, to wait before recovering a UNI port.

**Step 13** Check the **N-PE Pseudo-wire on SVI** check box to have ISC generate forwarding commands under SVIs (switch virtual interfaces).

By default, this check box is not checked. In this case, ISC generates forwarding commands under the service instance.

For a FlexUNI link, the attribute N-PE Pseudo-wire on SVI is dependent on the value of the attribute Configure with Bridge Domain (this is available in the policy workflow in the EVC Policy Editor - Service Options window). N-PE Pseudo-wire on SVI, if enabled, will be reflected only when Configure with Bridge Domain is set to true. Otherwise, the service request will not be created with xconnect under SVI, even if N-PE Pseudo-wire on SVI is enabled.

Usage notes:

- ISC supports a hybrid configuration for FlexUNI/EVC service requests. In a hybrid configuration, the forwarding commands (such as xconnect) for one side of an attachment circuit can be configured under a service instance, and the xconnect configuration for the other side of the attachment circuit can be configured under a switch virtual interface (SVI).
- For examples of these cases, see configlet examples FlexUNI/EVC (Pseudowire Core Connectivity, Bridge Domain, Pseudowire on SVI), page A-39 and FlexUNI/EVC (Pseudowire Core Connectivity, no Bridge Domain, no Pseudowire on SVI), page A-40.
- N-PE Pseudo-wire on SVI is applicable for all connectivity types, but a hybrid SVI configuration is possible only for pseudowire connectivity.
- When MPLS Core Connectivity Type is set as LOCAL connectivity type, the N-PE Pseudo-wire on SVI attribute is always disabled in the policy and service request.
- The N-PE Pseudo-wire on SVI attribute is not supported for IOS XR devices. All the xconnect commands are configured on L2 subinterfaces/service instance.
- Table 5-2 shows various use cases for hybrid configuration for FlexUNI/EVC service requests.

*Table 5-2 Use Cases for Hybrid Configuration for FlexUNI /EVC Service Requests*

| Use Bridge Domain | FlexUNI | N-PE Pseudowire on SVI | CLIs Generated |
|---|---|---|---|
| True | True | True | • xconnect under VLAN interface.<br>• Service instance under main interface. |
| True | True | False | • xconnect under service instance.<br>• Service instance under main interface. |
| False | True | N/A | • xconnect under service instance.<br>• Service instance under main interface. |
| True | False | True | xconnect under VLAN interface. |
| True | False | False | xconnect under subinterface. |
| False | False | False | xconnect under subinterface. |

**Step 14** Specify the type of **VLAN Translation** for this policy by clicking the appropriate radio button.

The choices are:

- **No**—No VLAN translation is performed. (This is the default.)
- **1:1**—1:1 VLAN translation.
- **2:1**—2:1 VLAN translation.

**Note** For detailed coverage of setting up VLAN translation, see Appendix C, “Setting Up VLAN Translation.”

**Note** VLAN translation is only supported on links that are specified as non-FlexUNI at the service request level.

**Step 15** Enter the **MTU Size** in bytes.

The maximum transmission unit (MTU) size is configurable and optional. The default size is 9216, and the range is 1500 to 9216. ISC does not perform an integrity check for this customized value. If a service request goes to the Failed Deploy state because this size is not accepted, you must adjust the size until the Service Request is deployed.

In ISC 6.0, different platforms support different ranges.

- For the 3750 and 3550 platforms, the MTU range is 1500 to 1546.
- For the Cisco 7600 Ethernet port, the MTU size is always 9216. Even with the same platform and same IOS release, different line cards support the MTU differently. For example, older line cards only take an MTU size of 9216 and newer cards support 1500 to 9216. However, ISC 6.0 uses 9216 in both cases.
- For the Cisco 7600 SVI (interface VLAN), the MTU size is 1500 to 9216.

**Step 16** Check the **PW Tunnel Selection** check box if you want to be able to manually select the Traffic Engineering (TE) tunnel for the pseudowire connecting point-to-point N-PEs.

This attribute is unchecked by default.

Subsequently, when you create a service request based on this policy, you must specify the TE tunnel ID in a field provided. ISC uses the tunnel information to create and provision a pseudowire class that describes the pseudowire connection between two N-PEs. This pseudowire class can be shared by more than one pseudowire, as long as the pseudowires share the same tunnel ID and remote loopback address. You are responsible to ensure that the tunnel interface and associated ID are configured. During service request creation when you specify the tunnel ID number, ISC does not check the validity of the value. That is, ISC does not verify the existence of the tunnel.

**Step 17** Check the **Use PseudoWireClass** check box to enable the selection of a pseudowire class.

This attribute is unchecked by default.

Usage notes:

- The pseudowire class name is used for provisioning pw-class commands on IOS XR devices. See Creating and Modifying Pseudowire Classes for IOS XR Devices, page 2-10 for additional information on pseudowire class support for IOS XR devices.
- If **Use PseudoWireClass** is checked, an additional attribute, **PseudoWireClass**, appears in the GUI. Click the **Select** button of PseudoWireClass attribute to choose a pseudowire class previously created in ISC.
- The Use PseudoWireClass attribute is only available if the MPLS core connectivity type was set as PSEUDOWIRE in the Service Options window (see Setting the Service Options, page 5-4).
- Use PseudoWireClass is only applicable for IOS XR devices.

**Step 18** For **L2VPN Group Name** choose one of the following from the drop-down list:

- **ISC**
- **VPNSC**

Usage notes:

- This attribute is used for provisioning the L2VPN group name on IOS XR devices.

**Note** The choices in the drop-down list are derived from a configurable DCPL property. For information about how to define the L2VPN Group Name choices available in the drop-down list, see Defining L2VPN Group Names for IOS XR Devices, page 2-14.

- L2VPN Group Name is only applicable for IOS XR devices.

**Step 19** Enter an **E-Line Name** to specify the point-to-point (p2p) E-line name.

Usage notes:

- If no value is specified for the **E-Line Name** in either the policy or the service request based on the policy, ISC autogenerates a default name as follows:
  - For PSEUDOWIRE core connectivity type, the format is:

    *DeviceName--VC_ID*
  - For LOCAL core connectivity type, the format is:

    *DeviceName--VLAN_ID*

  If the default name is more than 32 characters, the device names are truncated.
- E-Line Name is only applicable for IOS XR devices.

**Step 20** If you want to enable template association for this policy, click the **Next** button.

See the section Enabling Template Association, page 5-19 for information about this feature.

**Step 21** To save the FlexUNI/EVC policy, click **Finish**.

To create a service request based on a FlexUNI/EVC ATM-Ethernet Interworking policy, see Chapter 6, "Managing a FlexUNI/EVC ATM-Ethernet Interworking Service Request."

# Enabling Template Association

The ISC template feature gives you a means to download free-format CLIs to a device. If you enable templates, you can create templates and data files to download commands that are not currently supported by ISC.

**Step 1** To enable template association for the policy, click the **Next** button in EVC Policy Editor - Interface Attribute window (before clicking **Finish**).

The Template Association window appears. In this window, you can enable template support and, optionally, associate templates and data files with the policy. For instructions about associating templates with policies and how to use the features in this window, see Appendix B, "Working with Templates and Data Files".

**Step 2** When you have completed setting up templates and data files for the policy, click **Finish** in the Template Association window to close it and return to the Policy Editor window.

**Step 3** To save the FlexUNI/EVC ATM-Ethernet Interworking policy, click **Finish**.

To create a service request based on a FlexUNI/EVCATM-Ethernet Interworking policy, see Chapter 6, "Managing a FlexUNI/EVC ATM-Ethernet Interworking Service Request."

CHAPTER 6

# Managing a FlexUNI/EVC ATM-Ethernet Interworking Service Request

This chapter provides information on how to provision a FlexUNI/EVC ATM-Ethernet Interworking service request. It contains the following sections:

## Overview

A FlexUNI/EVC ATM-Ethernet Interworking service request allows you to configure interfaces on an N-PE to support the FlexUNI/EVC features described in Chapter 5, “Creating a FlexUNI/EVC ATM-Ethernet Interworking Policy.” To create a FlexUNI/EVC ATM-Ethernet Interworking service request, a FlexUNI/EVC ATM-Ethernet Interworking service policy must already be defined, as described in Chapter 5, “Creating a FlexUNI/EVC ATM-Ethernet Interworking Policy.” Based on the predefined FlexUNI/EVC policy, an operator creates a FlexUNI/EVC service request, with or without modifications to the policy, and deploys the service. One or more templates can also be associated to the N-PE as part of the service request.

ATM-Ethernet interworking is supported through the following configurations:

- ATM Transport Mode (VC)
  - ATM-Ethernet Pseudowire
  - ATM-ATM Local connect
  - ATM-Ethernet Local connect
- ATM Transport Mode (VP)
  - ATM-ATM Local connect

The following steps are involved in creating a FlexUNI/EVC ATM-Ethernet Interworking service request:

- Choose an existing FlexUNI/EVC ATM-Ethernet Interworking policy.

- Choose a VPN.

**Note** When working with VPN objects in the context of FlexUNI/EVC policies and service requests, only the VPN name and customer attributes are relevant. Other VPN attributes related to MPLS and VPLS are ignored.

- Specify a bridge domain configuration (if applicable).
- Specify a service request description.
- Specify automatic or manual allocation of the VC ID or VPLS VPN ID.
- Add direct connect links (if applicable).
- Add links with L2 access nodes (if applicable).
- Choose the N-PE and UNI interface for links.
- For links with L2 access nodes, choose a Named Physical Circuit (NPC) if more than one NPC exists from the N-PE or the UNI interface.
- Edit the link attributes.
- Modify the service request.
- Save the service request.

# Creating a FlexUNI/EVC ATM-Ethernet Interworking Service Request

To create a FlexUNI/EVC ATM-Ethernet Interworking service request, perform the following steps.

**Step 1** Choose **Service Inventory > Inventory and Connection Manager > Service Requests**.

The Service Requests window appears.

**Step 2** Click **Create**.

**Step 3** Choose **FlexUNI (EVC)** from the drop-down list.

The Select EVC Policy window appears. If more than one FlexUNI/EVC policy exists, a list of FlexUNI/EVC policies appears. FlexUNI/EVC service requests must be associated with a FlexUNI/EVC policy. You choose a FlexUNI/EVC ATM-Ethernet Interworking policy from the policies previously created (see Chapter 5, “Creating a FlexUNI/EVC ATM-Ethernet Interworking Policy”).

**Step 4** Choose a FlexUNI/EVC ATM-Ethernet Interworking policy from the list.

**Step 5** Click **OK**.

The EVC Service Request Editor window appears. The new service request inherits all the properties of the chosen FlexUNI/EVC ATM-Ethernet Interworking policy, such as all the editable and non-editable features and pre-set parameters.

**Step 6** Continue with the steps contained in the next section, Setting the Service Request Details, page 6-3.

# Setting the Service Request Details

After you have selected the FlexUNI/EVC policy to be used as the basis of the service request, the EVC Service Request Editor window appears. It is divided into three main sections:

- Service Request Details
- Direct Connect Links (no NPCs)
- Links with L2 Access Nodes (involves NPCs)

This window enables you to specify options for the service request, as well as configure directly connected links and links with L2 access nodes. The options displayed in first section of the window change, depending on the MPLS Core Connectivity Type that was specified in the policy (pseudowire or local). For clarity, each of these scenarios is presented in a separate section below, to highlight the different window configurations and behavior of the displayed options.

Proceed to the appropriate section, as determined by the MPLS Core Connectivity Type for the policy:

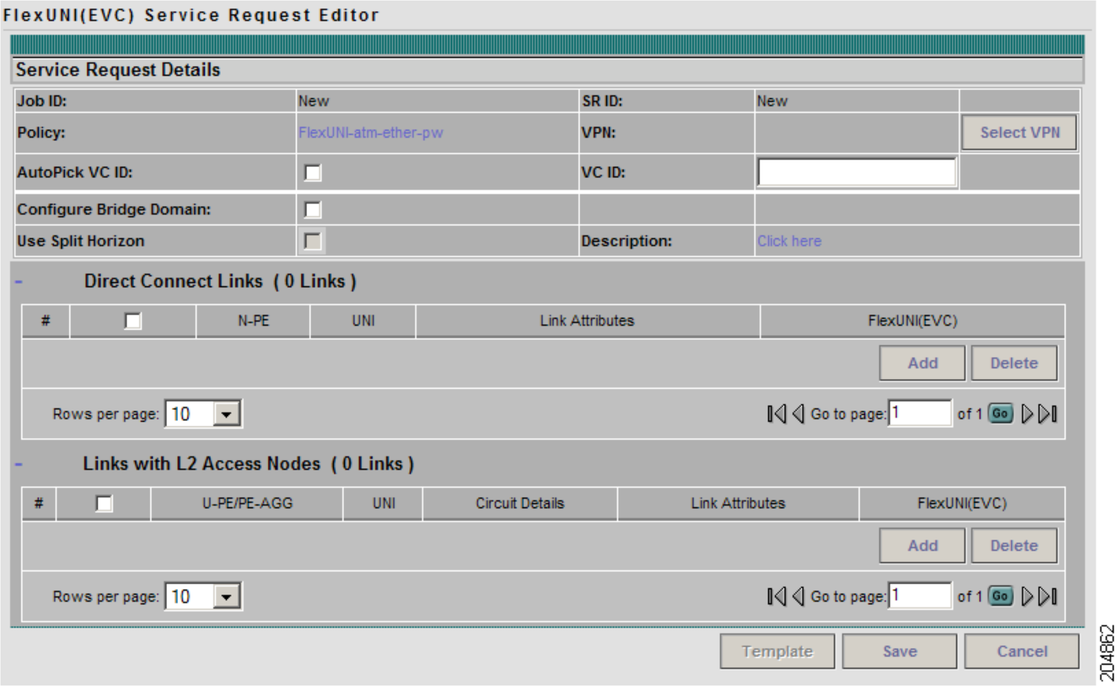
- Pseudowire Core Connectivity, page 6-3
- Local Core Connectivity, page 6-6

Instructions for setting up direct connect links and links with L2 access nodes are presented in later sections.

## Pseudowire Core Connectivity

If the MPLS Core Connectivity Type for the FlexUNI/EVC ATM-Ethernet Interworking policy is PSEUDOWIRE, the EVC Service Request Editor window shown Figure 6-1 appears.

*Figure 6-1 EVC Service Request Details Window for Pseudowire Core Connectivity*



| | | | |
|---|---|---|---|
| Job ID: | New | SR ID: | New |
| Policy: | FlexUNI-atm-ether-pw | VPN: | |
| AutoPick VC ID: | □ | VC ID: | |
| Configure Bridge Domain: | □ | | |
| Use Split Horizon | □ | Description: | Click here |

Direct Connect Links ( 0 Links )

| # | □ | N-PE | UNI | Link Attributes | FlexUNI(EVC) |
|---|---|---|---|---|---|

Rows per page: 10 Go to page: 1 of 1

Links with L2 Access Nodes ( 0 Links )

| # | □ | U-PE/PE-AGG | UNI | Circuit Details | Link Attributes | FlexUNI(EVC) |
|---|---|---|---|---|---|---|

Rows per page: 10 Go to page: 1 of 1

Perform the following steps to set the attributes in the first section of the Service Request Details window:

**Note** The **Job ID** and **SR ID** fields are read-only. When the service request is being created for the first time, the fields display a value of NEW. When an existing service request is being modified, the values of the fields indicate the respective IDs that the ISC database holds within the editing flow of the service request.

**Note** The **Policy** field is read-only. It displays the name of the policy on which the service request is based. Clicking on the read-only policy name displays a list of all the attribute values set within the policy.

**Step 1** Click **Select VPN** to choose a VPN for use with this service request.

The Select VPN window appears with the VPNs defined in the system.

**Note** The same VPN can be used by service requests with LOCAL and PSEUDOWIRE core types. If a VPN for a service request is used with VPLS core type, the same VPN cannot be used for service requests with LOCAL or PSEUDOWIRE core type.

**Step 2** Choose a **VPN Name** in the Select column.

**Step 3** Click **Select**.

The EVC Service Request Editor window appears with the VPN name displayed.

**Step 4** Check the **AutoPick VC ID** check box if you want ISC to choose a VC ID.

If you do not check this check box, you will be prompted to provide the ID in the VC ID field, as covered in the next step.

When AutoPick VC ID is checked, ISC allocates a VC ID for pseudowires from the ISC-managed VC ID resource pool. In this case, the text field for the VC ID option is non-editable.

**Step 5** If AutoPick VC ID was unchecked, enter a VC ID in the **VC ID** field.

Usage notes:

- The AutoPick VC ID attribute appears during the creation of a FlexUNI/EVC pseudowire service request.
- The VC ID value must be an integer value corresponding to a VC ID.
- When a VC ID is manually allocated, ISC verifies the VC ID to see if it lies within ISC's VC ID pool. If the VC ID is in the pool but not allocated, the VC ID is allocated to the service request. If the VC ID is in the pool and is already in use, ISC prompts you to allocate a different VC ID. If the VC ID lies outside of the ISC VC ID pool, ISC does not perform any verification about whether or not the VC ID allocated. The operator must ensure the VC ID is available.
- The VC ID can be entered only while creating a service. If you are editing the service request, the VC ID field is not editable.

**Step 6** Check the **Configure Bridge Domain** check box to determine bridge domain characteristics.

The behavior of the Configure Bridge Domain option works in tandem with the choice you selected in the MPLS Core Connectivity Type option in the FlexUNI/EVC policy, which in this case is pseudowire core connectivity. There are two cases:

- With FlexUNI:
  - If **Configure With Bridge Domain** is checked, the policy will configure pseudowires under SVIs associated to the bridge domain.
  - If **Configure With Bridge Domain** is unchecked, the policy will configure pseudowires directly under the service instance. This will conserve the global VLAN.
- Without FlexUNI:
  - If **Configure With Bridge Domain** is checked, the policy will configure pseudowires under SVIs.
  - If **Configure With Bridge Domain** is unchecked, the policy will configure pseudowires directly under subinterfaces.

  Pseudowires can be configured either directly under service instance of the corresponding FlexUNI-capable interface or under SVIs associated to the bridge domain.

**Step 7** Check the **Use Split Horizon** check box to enable split horizon with bridge domain.

Usage notes:

- The Use Split Horizon attribute is disabled by default.
- The Use Split Horizon attribute can be used only when the Configure Bridge Domain check box is checked (enabled).
- When Use Split Horizon is enabled, the **bridge domain** command in the CLI will be generated with split horizon. When it is disabled, the **bridge domain** command will be generated without split horizon.

**Step 8** Click the "Click here" link of the **Description** attribute to enter a description label for the service request.

This is useful for searching the ISC database for the particular service request.

A dialogue appears in which you can enter a description.

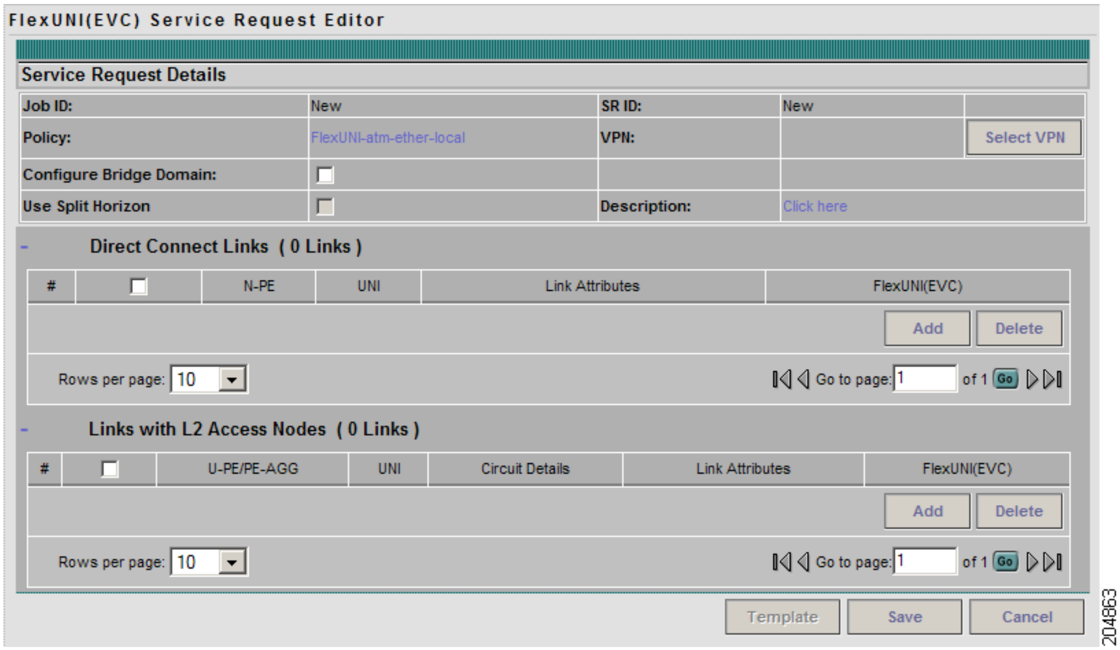
**Step 9** To set up direct connect links, see the section Setting Direct Connect Links, page 6-8.

**Step 10** To set up links with L2 access nodes, see the section Setting Links with L2 Access Nodes, page 6-22.

# Local Core Connectivity

If the MPLS Core Connectivity Type for the FlexUNI/EVC ATM-Ethernet Interworking policy is LOCAL, the EVC Service Request Editor window shown Figure 6-2 appears.

***Figure 6-2 EVC Service Request Details Window for Local Core Connectivity***



FlexUNI(EVC) Service Request Editor

Service Request Details

| | | | | |
|---|---|---|---|---|
| Job ID: | New | SR ID: | New | |
| Policy: | FlexUNI-atm-ether-local | VPN: | | Select VPN |
| Configure Bridge Domain: | □ | | | |
| Use Split Horizon | □ | Description: | Click here | |

- Direct Connect Links ( 0 Links )

| # | □ | N-PE | UNI | Link Attributes | FlexUNI(EVC) |
|---|---|---|---|---|---|

Add Delete

Rows per page: 10 Go to page: 1 of 1 Go

- Links with L2 Access Nodes ( 0 Links )

| # | □ | U-PE/PE-AGG | UNI | Circuit Details | Link Attributes | FlexUNI(EVC) |
|---|---|---|---|---|---|---|

Add Delete

Rows per page: 10 Go to page: 1 of 1 Go

Perform the following steps to set the attributes in the first section of the Service Request Details window:

**Step 1** The **Job ID** and **SR ID** fields are read-only.

When the service request is being created for the first time, the fields display a value of NEW. When an existing service request is being modified, the values of the fields indicate the respective IDs that the ISC database holds within the editing flow of the service request.

**Step 2** The **Policy** field is read-only.

It displays the name of the policy on which the service request is based.

**Step 3** Click **Select VPN** to choose a VPN for use with this service request.

The Select VPN window appears with the VPNs defined in the system.

**Note** The same VPN can be used by service requests with LOCAL and PSEUDOWIRE core types. If a VPN for a service request is used with VPLS core type, the same VPN cannot be used for service requests with LOCAL or PSEUDOWIRE core type.

**Step 4** Choose a **VPN Name** in the Select column.

**Step 5** Click **Select**.

The EVC Service Request Editor window appears with the VPN name displayed.

**Step 6** Check the **Configure Bridge Domain** check box to determine bridge domain characteristics.

Usage notes:

- If Configure Bridge Domain is checked, all links will have the same bridge domain ID allocated from the VLAN pool on the N-PE. All non-FlexUNI links will have the Service Provider VLAN as the bridge domain ID. On the other hand, if no FlexUNI links are added, the Service Provider VLAN will be allocated first and this will be used as the bridge domain ID when FlexUNI links are added.
- If Configure Bridge Domain is unchecked, a maximum of two links that terminate on the same N-PE can be added. (This uses the **connect** command available in the EVC infrastructure.) This is only supported for ATM-ATM local connect.

  **Note** See the following comments for details on how ISC autogenerates the connect name.

  Because the device only accepts a maximum of 15 characters for the connect name, the connect name is generated using the following format:

  *CustomerNameTruncatedToMaxPossibleCharacters_ServiceRequestJobID*

  For example, if the customer name is NorthAmericanCustomer and the service request job ID is 56345, the autogenerated connect name would be NorthAmer_56345.

  The CLI generated would be:

  ```
  connect NorthAmer_56345 ATM7/0/5 11 ATM7/0/4 18
  ```

  In this case, 11 and 18 are service instance VPIs.
- If the policy setting for Configure Bridge Domain is non-editable, the option in the service request will be read-only.

**Step 7** Check the **Use Split Horizon** check box to enable split horizon with bridge domain.

Usage notes:

- The Use Split Horizon attribute is disabled by default.
- The Use Split Horizon attribute can be used only when the Configure Bridge Domain check box is checked (enabled).
- When Use Split Horizon is enabled, the **bridge domain** command in the CLI will be generated with split horizon. When it is disabled, the **bridge domain** command will be generated without split horizon.

**Step 8** Click the "Click here" link of the **Description** attribute to enter a description label for the service request.

A dialogue appears in which you can enter a description.

**Step 9** To set up direct connect links, see the section Setting Direct Connect Links, page 6-8.

**Step 10** To set up links with L2 access nodes, see the section Setting Links with L2 Access Nodes, page 6-22.

# Setting up Links to the N-PE

The lower two sections of the EVC Service Request Editor window allow you to set up links to the N-PE. For direct connect links, the CE is directly connected to the N-PE, with no intermediate L2 access nodes. For links with L2 access nodes, there are intermediate devices present between the CE and NPE requiring NPCs to be created in ISC.

The Direct Connect Link section of the window is where you set up links that directly connect to the N-PE. No NPCs are involved. ATM links are supported for direct connect links.

The Links with L2 Access Nodes section is where you set up links with L2 (Ethernet) access nodes. NPCs are involved.

**Note** ATM interfaces cannot be in L2 access nodes.

See the appropriate section, depending on which type of link you are setting up:

- Setting Direct Connect Links, page 6-8
- Setting Links with L2 Access Nodes, page 6-22

**Note** Many of steps for setting up the two link types are the same. The basic workflow for setting up links, as well as the attributes to be set, are presented in the following section Setting Direct Connect Links, page 6-8. Even if you are setting up links with L2 access nodes, it will be helpful to refer to the information presented in that section, as the section on L2 access nodes only covers the unique steps for such links.

## Setting Direct Connect Links

Perform the following steps to set up the direct connect links. Most of these steps apply to links with L2 access nodes also.

**Step 1** Click **Add** to add a link.

A new numbered row for the link attributes appears.

**Step 2** Click **Select NPE** in N-PE column.

The Select PE Device window appears. This window displays the list of currently defined PEs.

a. The **Show PEs with** drop-down list shows PEs by Provider, PE Region Name, or by Device Name.

b. The **Find** button allows a search for a specific PE or a refresh of the window.

c. The **Rows per page** drop-down list allows the user to configure the number of entries displayed on the screen at one time.

**Step 3** In the **Select** column, choose the PE device name for the link.

**Step 4** Click **Select**.

The EVC Service Request Editor window reappears displaying the name of the selected PE in the NPE column.

**Step 5** Choose the UNI interface from the drop-down list in the UNI column.

**Note** ISC only displays the available interfaces for the service, based on the configuration of the underlying interfaces, existing service requests that might be using the interface, and the customer associated with the service request. You can click the **Details** button to display a pop-up window with information on the available interfaces, such as interface name, customer name, VPN name, job ID, service request ID, service request type, translation type, and VLAN ID information.

**Note** When the UNI is configured on an N-PE device running IOS XR, the Standard UNI Port attribute is not supported. All the CLIs related to Standard UNI Port and UNI Port Security are ignored in this case.

**Step 6** Check the **FlexUNI** check box to mark the link for configuring service instance for the links.

**Note** The FlexUNI check box is mentioned at this stage because the setting of the check box alters the behavior of the link editing function available in the Link Attributes column. This is covered in the next steps.

### Editing the Link Attributes

The next steps document the use of the **Edit** link in the Link Attributes column. (In the case where the link attributes have already been set, this link changes from **Edit** to **Change**.) The link editing workflow changes depending on the status of the FlexUNI check box for the link. If the FlexUNI check box is checked, the editing workflow involves setting attributes in two windows, for two sets of link attributes:

- The FlexUNI Details
- Standard UNI Details

If the FlexUNI check box for the link is not checked, only the Standard UNI Details window is presented.

In the steps that follow, both scenarios covered.

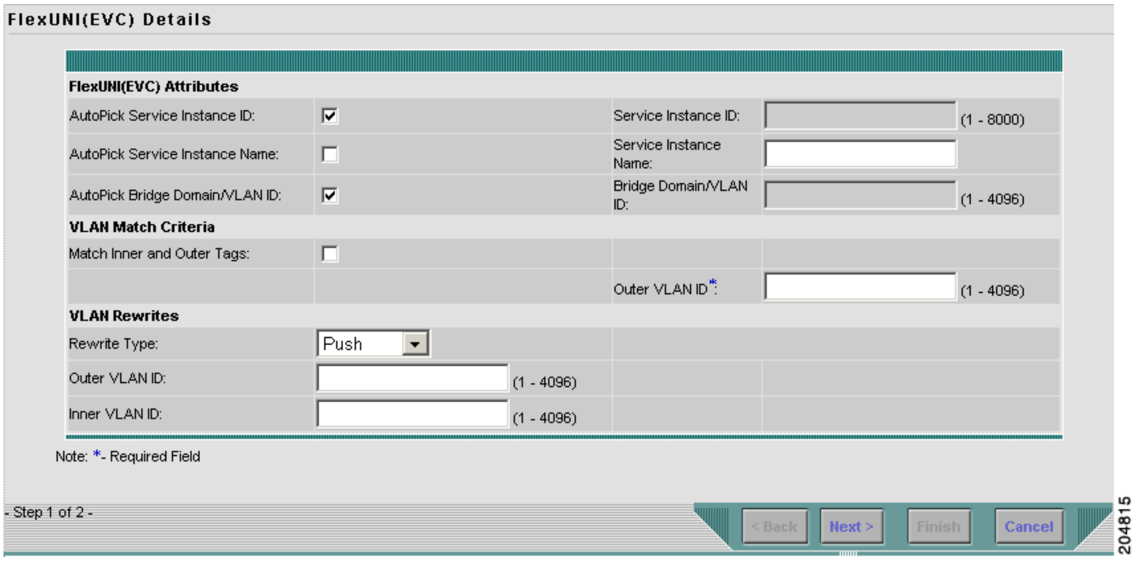
**Note** If you are setting up and ATM link (by choosing an ATM interface as the UNI on the N-PE device, there is a different workflow. The check box in the FlexUNI(EVC) column dynamically disappears, and clicking the Edit link in the link attributes column brings up the ATM-Ethernet Attributes window. For information on using this window to set up an ATM link, see Setting the ATM Link Attributes, page 6-18.

**Step 7** Click **Edit** in the Link Attributes column to specify the UNI attributes.

### FlexUNI Details Window

If the FlexUNI check box is checked, the FlexUNI Details window appears, as shown in Figure 6-3.

*Figure 6-3 FlexUNI Details Window*



FlexUNI(EVC) Details

| | | | |
|---|---|---|---|
| **FlexUNI(EVC) Attributes** | | | |
| AutoPick Service Instance ID: | ☑ | Service Instance ID: | (1 - 8000) |
| AutoPick Service Instance Name: | ☐ | Service Instance Name: | |
| AutoPick Bridge Domain/VLAN ID: | ☑ | Bridge Domain/VLAN ID: | (1 - 4096) |
| **VLAN Match Criteria** | | | |
| Match Inner and Outer Tags: | ☐ | | |
| | | Outer VLAN ID*: | (1 - 4096) |
| **VLAN Rewrites** | | | |
| Rewrite Type: | Push | | |
| Outer VLAN ID: | (1 - 4096) | | |
| Inner VLAN ID: | (1 - 4096) | | |

Note: *- Required Field

- Step 1 of 2 -

All of the fields in the FlexUNI Details screen are enabled based on the policy settings. For example, if Both Tags is selected in the policy and is editable, then the Match Inner and Outer Tags check box will be selected and editable in this window. The behavior is similar for the other attributes in the FlexUNI Details window

**Step 8** Check the **AutoPick Service Instance ID** check box to specify that the service instance ID will be autogenerated and allocated to the link during service request creation.

If the check box is unchecked, you must specify the service instance ID (see the next step).

Usage notes:

- The service instance ID represents an Ethernet Flow Point (EFP) on an interface in the EVC infrastructure. The service instance ID is locally significant to the interface. This ID has to be unique only at the interface level. The ID must be a value from 1 to 8000.
- There are no resource pools available in ISC from which to allocate the service instance IDs.
- In the case of a manually provided service instance ID, it is the responsibility of the operator to maintain the uniqueness of the ID at the interface level.
- This attribute is not displayed for IOS XR devices.

**Step 9** If the AutoPick Service Instance ID check box is not checked, enter an appropriate value for the service instance ID in the **Service Instance ID** field.

**Step 10** Check the **AutoPick Service Instance Name** check box to specify that the service instance name will be autogenerated.

If the check box is unchecked, you can specify the service instance name (see the next step).

Usage notes:

- If the check box is checked, the Service Instance Name text field is disabled.
- The service instance name is autogenerated in the following pattern: *CustomerName_ServiceRequestJobID*.

- For example configlets, see FlexUNI/EVC (No AutoPick Service Instance Name, No Service Instance Name), page A-42, FlexUNI/EVC (User-Provided Service Instance Name, Pseudowire Core Connectivity), page A-43, and FlexUNI/EVC (User-Provided Service Instance Name, Local Core Connectivity), page A-44.
- This attribute is not displayed for IOS XR devices.

**Step 11** If the AutoPick Service Instance Name check box is not checked, enter an appropriate value for the service instance ID in the **Service Instance Name** field.

Usage notes:

- The text string representing the service instance name must be 40 characters or less and contain no spaces. Other special characters are allowed.
- If AutoPick Service Instance Name is unchecked and no service instance name is entered in the text field, then ISC does not generate the global **ethernet evc evcname** command in the device configuration generated by the service request.

**Step 12** Check the **AutoPick Bridge Domain/VLAN ID** check box to have ISC autopick the VLAN ID for the service request during service request creation.

If this check box is unchecked, the you must specify a bridge domain VLAN ID (see the next step).

Usage notes:

- AutoPick Bridge Domain/VLAN ID consumes a global VLAN ID on the device.
- The bridge domain VLAN ID is picked from the existing ISC VLAN pool.
- This attribute is not displayed for IOS XR devices.

**Step 13** If the AutoPick Bridge Domain/VLAN ID check box is unchecked, enter an appropriate value in the **Bridge Domain/VLAN ID** field.

**Note** This configuration applies in conjunction with the Configure Bridge Domain option in the EVC Service Request Editor window. If the option is not enabled in that window, then AutoPick Bridge Domain/VLAN ID check box is redundant and not required.

When a VLAN ID is manually allocated, ISC verifies the VLAN ID to see if it lies within ISC's VLAN ID pool. If the VLAN ID is in the pool but not allocated, the VLAN ID is allocated to the service request. If the VLAN ID is in the pool and is already in use, ISC prompts you to allocate a different VLAN ID. If the VLAN ID lies outside of the ISC VLAN ID pool, ISC does not perform any verification about whether the VLAN ID allocated. The operator must ensure the VLAN ID is available.

**Step 14** Check the **Match Inner and Outer Tags** check box to enable service requests created with the policy to match both the inner and outer VLAN tags of the incoming frames.

If you do not check this check box, service requests created with the policy will match only the outer VLAN tag of the incoming frames.

Checking the Match Inner and Outer Tags attribute causes the Inner VLAN ID and Outer VLAN ID fields (covered in the next steps) to appear.

**Step 15** If the Match Inner and Outer Tags check box is checked, enter the inner and outer VLAN tags in the **Inner VLAN ID** and **Outer VLAN ID** fields.

Usage notes:

- You can specify single values, single ranges, multiples values, multiple ranges, or combinations of these. Examples:
  - 10

- 10, 15,17
- 10-15
- 10-15,17-20
- 10,20-25

- If the Inner VLAN Ranges attribute is set to true in the policy, the Inner VLAN ID field can take a range of inner VLAN tags.

**Step 16** If the Match Inner and Outer Tags check box is unchecked, enter the outer VLAN tag in the **Outer VLAN ID** field.

**Note** The VLAN specified in Outer VLAN ID will be provisioned on the rest of the L2 access nodes (if the link has any), including the customer-facing UNI.

**Step 17** In the VLAN Rewrite section of the window, choose a **Rewrite Type** from the drop-down list.

The choices are:

- **Pop**
- **Push**
- **Translate**

The subsequent attributes in the GUI change depending on the choice of Rewrite Type, as covered in the next steps.

**Step 18** If Pop is the Rewrite Type, two check boxes appear:

a. Check the **Pop Outer Tag** check box to pop the outer VLAN ID tag of the incoming frames that fulfill the match criteria. If this check box is unchecked, the outer tag of the incoming traffic will not be popped.

b. Check the **Pop Inner Tag** check box to pop the inner VLAN ID tag of the incoming frames that fulfill the match-criteria. If this check box is unchecked, the inner tag will not be changed.

Note that if Pop Inner Tag is checked, Pop Outer Tag is automatically checked.

**Step 19** If Push is the Rewrite Type, two text boxes appear:

a. In the text box **Outer VLAN ID**, enter an outer VLAN ID tag that will be imposed on the incoming frames that fulfill the match criteria. All service requests created with this setting push a dot1q outer tag on the incoming frames matching the match criteria. If a value is not provided, the push operation is ignored and not configured on the device.

b. In the text box **Inner VLAN ID**, enter an inner VLAN ID tag that will be imposed on the incoming frames that fulfill the match criteria. All service requests created with this setting push a dot1q inner tag on the incoming frames matching the match criteria. The Inner VLAN tag cannot be pushed without an Outer VLAN tag. That is, when pushing an Inner VLAN tag, the Outer VLAN tag also must be defined.

**Step 20** If Translate is the Rewrite Type, a **Translation Type** drop-down list appears.

The choices available in this list vary depending on the setting of the Match Inner and Outer Tags attribute (set in a previous step).

a. If the Match Inner and Outer Tags check box is checked (true), choose a translation type of **1:1**, **1:2**, **2:1**, or **2:2** from the Translation Type drop-down list.

- If you choose 1:1 or 2:1, enter a value in the **Outer VLAN ID** text box that appears. The outer tag of all the incoming frames that fulfill the match criteria will be translated to this ID.

- If you choose 1:2 or 2:2, enter values in the **Outer VLAN ID** and **Inner VLAN ID** text boxes that appear. The outer and inner tags of all the incoming frames that fulfill the match criteria will be translated to these IDs.

**b.** If the Match Inner and Outer Tags check box is unchecked (false), choose a translation type of **1:1** or **1:2** from the Translation Type drop-down list.

- If you choose 1:1, enter a value in the **Outer VLAN ID** text box that appears. The outer tag of all the incoming frames that fulfill the match criteria will be translated to this ID.
- If you choose 1:2, enter values in the **Outer VLAN ID** and **Inner VLAN ID** text boxes that appear. The outer and inner tags of all the incoming frames that fulfill the match criteria will be translated to these IDs.

**Step 21** Clicked **Next** to save the settings in the FlexUNI Details window.

The Standard UNI Details window appears, as shown in Figure 6-5.

**Step 22** Continue with setting the standard UNI link attributes in the next steps.

### Editing the Standard UNI Attributes

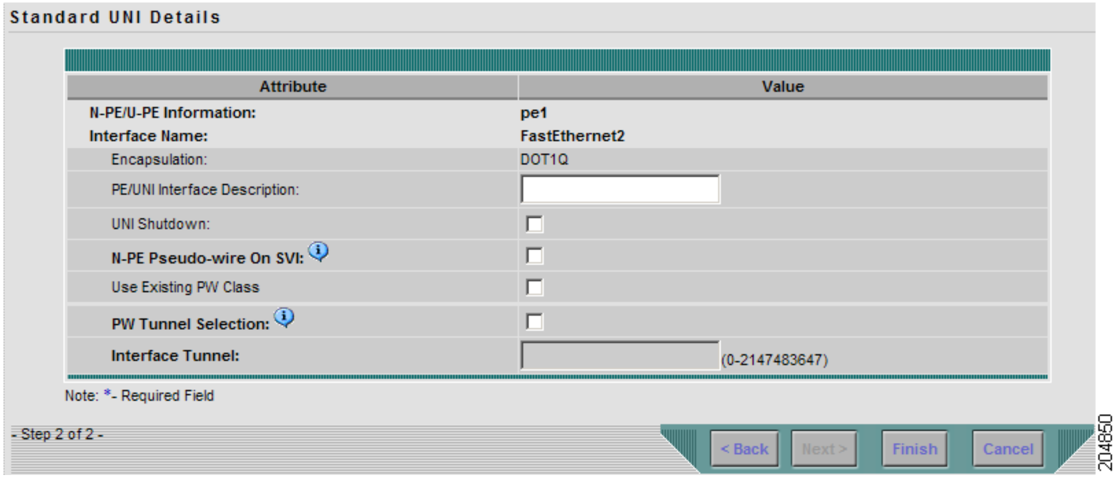
The following steps cover setting the attributes in the Standard UNI Details window. In the case of a link which is not set as a FlexUNI link (by not checking the FlexUNI check box in the Service Request Details window), editing the link attributes begins with this window.

**Note** The attributes that appear in the Standard UNI Details window are dynamically configured by ISC. Some of the attributes covered in the steps below might not appear in the window, depending on the policy and service request settings or the link type. For example, if the MPLS core connectivity type of the FlexUNI/EVC policy is local, the pseudowire-related attributes will not appear. Also, setting the link as FlexUNI or non-FlexUNI will change the attributes that appear in the window. In addition, attributes are filtered based on device type (IOS or IOS XR). These cases are noted in the steps, for reference.

Some possible presentations of the Standard UNI attributes are shown in Figure 6-4 through Figure 6-6.

Figure 6-4 is an example of a direct connect link for a Cisco 7600 device running IOS with pseudowire core connectivity, configure bridge domain enabled, and the FlexUNI check box checked.

***Figure 6-4 Standard UNI Details Window (Cisco 7600, Configure Bridge Domain Enabled)***



Standard UNI Details

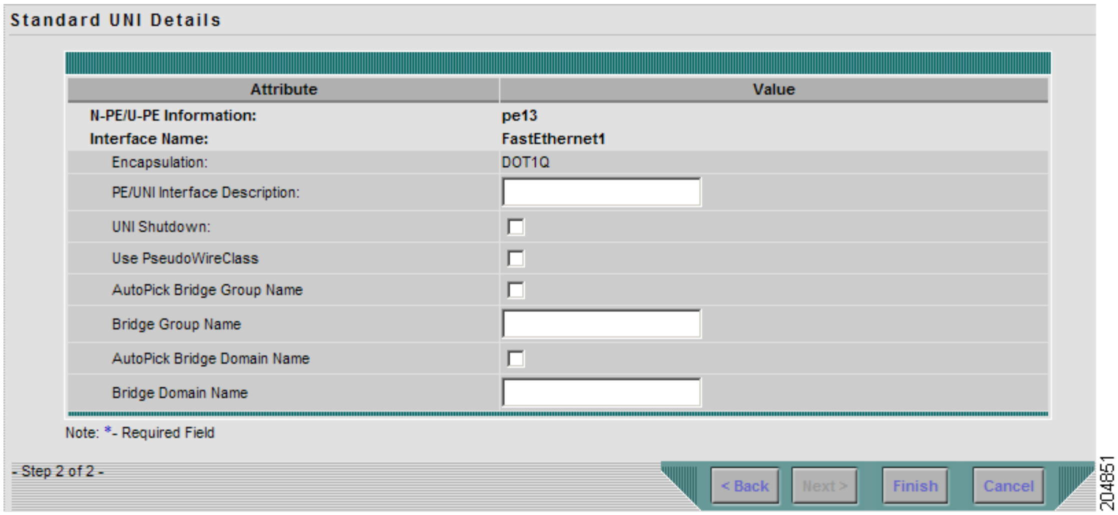
| Attribute | Value |
|---|---|
| N-PE/U-PE Information: | pe1 |
| Interface Name: | FastEthernet2 |
| Encapsulation: | DOT1Q |
| PE/UNI Interface Description: | |
| UNI Shutdown: | □ |
| N-PE Pseudo-wire On SVI: ⓘ | □ |
| Use Existing PW Class | □ |
| PW Tunnel Selection: ⓘ | □ |
| Interface Tunnel: | (0-2147483647) |

Note: *- Required Field

- Step 2 of 2 -

Figure 6-5 is an example of a direct connect link for a Cisco ASR 9000 running IOS XR with pseudowire core connectivity, configure bridge domain enabled, and the FlexUNI check box checked.

*Figure 6-5 Standard UNI Details Window (Cisco ASR 9000, Configure Bridge Domain Enabled)*



Standard UNI Details

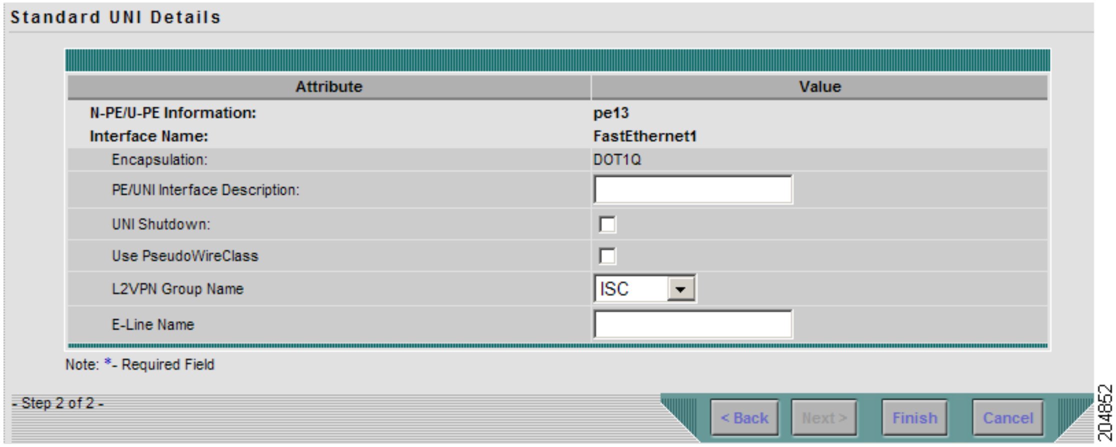
| Attribute | Value |
|---|---|
| N-PE/U-PE Information: | pe13 |
| Interface Name: | FastEthernet1 |
| Encapsulation: | DOT1Q |
| PE/UNI Interface Description: | |
| UNI Shutdown: | □ |
| Use PseudoWireClass | □ |
| AutoPick Bridge Group Name | □ |
| Bridge Group Name | |
| AutoPick Bridge Domain Name | □ |
| Bridge Domain Name | |

Note: *- Required Field

- Step 2 of 2 -

Figure 6-6 is an example of a direct connect link for a Cisco ASR 9000 running IOS XR with pseudowire core connectivity, configure bridge domain not enabled, and the FlexUNI check box checked.

*Figure 6-6 Standard UNI Details Window (Cisco ASR 9000, Configure Bridge Domain not Enabled)*



Standard UNI Details

| Attribute | Value |
|---|---|
| N-PE/U-PE Information: | pe13 |
| Interface Name: | FastEthernet1 |
| Encapsulation: | DOT1Q |
| PE/UNI Interface Description: | |
| UNI Shutdown: | □ |
| Use PseudoWireClass | □ |
| L2VPN Group Name | ISC |
| E-Line Name | |

Note: *- Required Field

- Step 2 of 2 -

**Step 23** The **N-PE/U-PE Information** and **Interface Name** fields display the PE device and interface name selected in previous steps.

These fields are read-only.

**Step 24** Choose an **Encapsulation** type from the drop-down list.

The choices are:

- **DOT1QTRUNK**—Configures the UNI as a trunk with 802.1q encapsulation. If the UNI belongs to a directly connected and FlexUNI link, this setting signifies that the incoming frames are 802.1q encapsulated and that they match the VLAN ID configured for the link. This specific topology does not involve a trunk UNI as such.
- **DOT1QTUNNEL**—Configures the UNI as an 802.1q tunnel (also known as a dot1q tunnel or Q-in-Q) port.
- **ACCESS**—Configures the UNI as an access port.

This attribute allows you to deploy different types of UNI encapsulation on different links of a service.

**Step 25** In the **PE/UNI Interface Description** field, enter a description for the interface, if desired.

**Step 26** Check the **UNI Shutdown** check box if you want to leave the UNI port shut during service activation (for example, when the service provider wants to deploy a service in the network but wants to activate it at a later time).

**Step 27** Specify the type of **VLAN Translation** for the service request by clicking the appropriate radio button.

The choices are:

- **No**—No VLAN translation is performed. (This is the default.)
- **1:1**—1:1 VLAN translation.
- **2:1**—2:1 VLAN translation.

**Note** VLAN translation, and all standard UNI and port security attributes are applicable for links with L2 access. If the UNI is on an N-PE, these attributes will not appear.

**Note** For detailed coverage of setting up VLAN translation, see Appendix C, “Setting Up VLAN Translation.”

**Step 28** Check the **N-PE Pseudo-wire on SVI** check box to have ISC generate forwarding commands under SVIs (switch virtual interfaces).

By default, this check box is not checked. In this case, ISC generates forwarding commands under the service instance.

For a FlexUNI link, the attribute N-PE Pseudo-wire on SVI is dependent on the value of the attribute Configure with Bridge Domain (this is available in the service request workflow in the EVC Service Request Editor window). N-PE Pseudo-wire on SVI, if enabled, will be reflected only when Configure with Bridge Domain is set to true. Otherwise, the service request will not be created with xconnect under SVI, even if N-PE Pseudo-wire on SVI is enabled.

Usage notes:

- For a FlexUNI link, the attribute N-PE Pseudo-wire on SVI is dependent on the value of the attribute Configure with Bridge Domain (in the EVC Service Request Editor window). N-PE Pseudo-wire on SVI, if enabled, will be reflected only when Configure with Bridge Domain is set to true. Otherwise, the service request will not be created with scanned under SVI, even if N-PE pseudo-wire on SVI is enabled.
- ISC supports a hybrid configuration for FlexUNI/EVC service requests. In a hybrid configuration, the forwarding commands (such as scanned) for one side of an attachment circuit can be configured under a service instance, and the xconnect configuration for the other side of the attachment circuit can be configured under a switch virtual interface (SVI).

- N-PE Pseudo-wire on SVI is applicable for all connectivity types (PSEUDOWIRE or LOCAL), but a hybrid SVI configuration is possible only for pseudowire connectivity.
- When MPLS Core Connectivity Type is set as LOCAL connectivity type, the N-PE Pseudo-wire on SVI attribute is always disabled in the policy and service request.
- For examples of these cases, see configlet examples FlexUNI/EVC (Pseudowire Core Connectivity, Bridge Domain, Pseudowire on SVI) and FlexUNI/EVC (Pseudowire Core Connectivity, no Bridge Domain, no Pseudowire on SVI).
- For additional information on the N-PE Pseudo-wire on SVI attribute, see the corresponding coverage in the FlexUNI/EVC policy chapter in the section Setting the Interface Attributes, page 3-16.
- The N-PE Pseudo-wire on SVI attribute is not supported for IOS XR devices. All the xconnect commands are configured on L2 subinterfaces/service instance.

**Step 29** Check the **Use Existing PW Class** check box to enable the selection of a pseudowire class.

This attribute is unchecked by default.

Usage notes:

- If Use Existing PW Class is checked, an additional attribute, **Existing PW Class Name**, appears in the GUI. Enter the name of a pseudowire class which already exists in the device.
- If Use Existing PW Class is checked, the PW Tunnel Selection and Interface Tunnel attributes will disappear from the window. This is to prevent ISC from generating the pseudowire class.
- The Use PseudoWireClass attribute is only available if the MPLS core connectivity type was set as PSEUDOWIRE in the Service Options window (see Pseudowire Core Connectivity, page 6-3).
- Use PseudoWireClass is only applicable for IOS XR devices.

**Step 30** Check the **PW Tunnel Selection** check box if you want to be able to manually select the Traffic Engineering (TE) tunnel for the pseudowire connecting point-to-point N-PEs.

Usage notes:

- Checking the PW Tunnel Selection check box activates the Interface Tunnel attribute field (see the next step).
- This attribute only appears if the MPLS core connectivity type is set as pseudowire in the FlexUNI/EVC policy.

**Step 31** If you checked the PW Tunnel Selection check box, enter the TE tunnel ID in the **Interface Tunnel** text field.

ISC uses the tunnel information to create and provision a pseudowire class that describes the pseudowire connection between two N-PEs. This pseudowire class can be shared by more than one pseudowire, as long as the pseudowires share the same tunnel ID and remote loopback address. During service request creation, ISC does not check the validity of the tunnel ID number. That is, ISC does not verify the existence of the tunnel.

**Step 32** Check the **Use PseudoWireClass** check box to enable the selection of a pseudowire class.

This attribute is unchecked by default.

Usage notes:

- The pseudowire class name is used for provisioning pw-class commands on IOS XR devices. See Creating and Modifying Pseudowire Classes for IOS XR Devices, page 2-10 for additional information on pseudowire class support for IOS XR devices.

- If Use PseudoWireClass is checked, an additional attribute, **PseudoWireClass**, appears in the GUI. Click the **Select** button of PseudoWireClass attribute to choose a pseudowire class previously created in ISC.
- The Use PseudoWireClass attribute is only available if the MPLS core connectivity type was set as PSEUDOWIRE in the Service Options window (see Setting the Service Options, page 5-4).
- Use PseudoWireClass is only applicable for IOS XR devices.

**Step 33** Check the **AutoPick Bridge Group Name** check box to have ISC autopick the bridge group name during service request creation.

If this check box is unchecked, you are prompted to specify a bridge group name during service request creation (see the next step).

Usage notes:

- This attribute only displays for IOS XR devices.
- If the AutoPick Bridge Group Name check box is unchecked, enter an bridge group name in the **Bridge Group Name** text field.

**Step 34** Check the **AutoPick Bridge Domain/VLAN ID** check box to have ISC autopick the VLAN ID during service request creation.

If this check box is unchecked, you are prompted to specify a VLAN ID during service request creation (see the next step).

Usage notes:

- AutoPick Bridge Domain/VLAN ID consumes a global VLAN ID on the device.
- The bridge domain VLAN ID is picked from the existing ISC VLAN pool.
- The AutoPick Bridge Domain/VLAN ID attribute appears for both Cisco 7600 and ASR 9000 devices. It will be displayed only for non-FlexUNI links.

**Step 35** If the AutoPick Bridge Domain/VLAN ID check box is unchecked, enter an ID number in the **Bridge Domain/VLAN ID** text field.

Usage notes:

- If AutoPick Bridge Domain/VLAN ID is checked, this field is non-editable.
- When a VLAN ID is manually allocated, ISC verifies the VLAN ID to see if it lies within ISC's VLAN ID pool. If the VLAN ID is in the pool but not allocated, the VLAN ID is allocated to the service request. If the VLAN ID is in the pool and is already in use, ISC prompts you to allocate a different VLAN ID. If the VLAN ID lies outside of the ISC VLAN ID pool, ISC does not perform any verification about whether the VLAN ID allocated. The operator must ensure the VLAN ID is available.
- The Bridge Domain/VLAN ID text field appears for both Cisco 7600 and ASR 9000 devices. It will be displayed only for non-FlexUNI links.

**Step 36** For **L2VPN Group Name** choose one of the following from the drop-down list:

- **ISC**
- **VPNSC**

Usage notes:

- This attribute is used for provisioning the L2VPN group name on IOS XR devices.

**Note** The choices in the drop-down list are derived from a configurable DCPL property. For information about how to define the L2VPN Group Name choices available in the drop-down list, see Defining L2VPN Group Names for IOS XR Devices, page 2-14.

- L2VPN Group Name is only applicable for IOS XR devices.

**Step 37** Enter an **E-Line Name** to specify the point-to-point (p2p) E-line name.

Usage notes:

- If no value is specified for the **E-Line Name**, ISC autogenerates a default name as follows:
  - For PSEUDOWIRE core connectivity type, the format is:

    *DeviceName--VC_ID*
  - For LOCAL core connectivity type, the format is:

    *DeviceName--VLAN_ID*

  If the default name is more than 32 characters, the device names are truncated.
- E-Line Name is only applicable for IOS XR devices.

**Step 38** Click **OK** to save the Standard UNI settings and return to the EVC Service Request window.

The value in the Link Attributes column now displays as “Changed,” signifying that the link settings have been updated. You can edit the link attributes now or at a future time by clicking on the Changed link and modifying the settings in the Standard UNI Details window.

See Modifying the FlexUNI/EVC Service Request, page 6-23 for details on editing the link attributes.

**Step 39** To add another link click the **Add** button and set the attributes for the new link as in the previous steps in this section.

**Step 40** To delete a link, check the check box in the first column of the row for that link and click the **Delete** button.

**Step 41** If you want to set up links with L2 access nodes for this service request, see Setting Links with L2 Access Nodes, page 6-22.

**Step 42** When you have completed setting the attributes in the EVC Service Request Editor window, click the **Save** button at the bottom of the window to save the settings and create the FlexUNI/EVC service request.

If any attributes are missing or incorrectly set, ISC displays a warning in the lower left of the window. Make any corrections or updates needed (based on the information provided by ISC), and click the **Save** button.

For information on modifying a FlexUNI/EVC service request see the section Modifying the FlexUNI/EVC Service Request, page 6-23. For additional information about saving a FlexUNI/EVC service request, see Saving the FlexUNI/EVC Service Request, page 6-24.

## Setting the ATM Link Attributes

This section describes how to set up a direct connect link as an ATM link.

Perform the following steps to set up the ATM link.

**Step 1** In the Direct Connect Links section of the FlexUNI(EVC) Service Request Editor window, specify the device for which you would like to set up an ATM link.

**Step 2** Choose an ATM interface for the UNI.

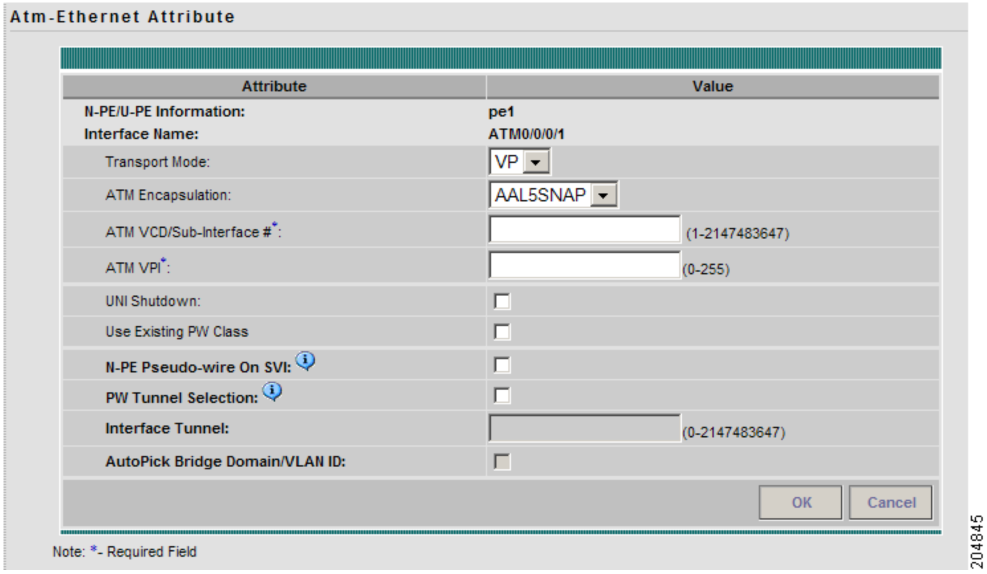
**Note** ATM interfaces are displayed in the drop-down list in the UNI column only if the FlexUNI/EVC service request is based on an ATM-Ethernet Interworking policy type.

When you choose an ATM interface, the check box in the FlexUNI(EVC) column dynamically disappears from the GUI.

**Step 3** In the Link Attributes column, click the **Edit** link of the device on which you want to add an ATM link.

The ATM-Ethernet Attributes window appears, as shown in Figure 6-7.

***Figure 6-7 ATM-Ethernet Attributes Window***



Atm-Ethernet Attribute

| Attribute | Value |
|---|---|
| N-PE/U-PE Information: | pe1 |
| Interface Name: | ATM0/0/0/1 |
| Transport Mode: | VP |
| ATM Encapsulation: | AAL5SNAP |
| ATM VCD/Sub-Interface #*: | (1-2147483647) |
| ATM VPI*: | (0-255) |
| UNI Shutdown: | □ |
| Use Existing PW Class | □ |
| N-PE Pseudo-wire On SVI: | □ |
| PW Tunnel Selection: | □ |
| Interface Tunnel: | (0-2147483647) |
| AutoPick Bridge Domain/VLAN ID: | □ |

Note: *- Required Field

All of the fields in the ATM-Ethernet Attributes window are enabled based on the policy settings.

**Step 4** Choose the **Transport Mode** from the drop-down list.

The choices are:

- **VP**—Virtual path mode. This is the default.
- **VC**—Virtual circuit mode.

**Step 5** Choose the **ATM Encapsulation** from the drop-down list.

- **AAL5SNAP**

**Step 6** To specify the ATM virtual channel descriptor (VCD)/subinterface number, enter a value in the **ATM VCD/Sub-Interface #** field.

The value can be from 1 to 2147483647.

**Step 7** To specify the ATM virtual path identifier (VPI), enter a value in the **ATM VPI** field.

The value can be from 0 to 255.

**Step 8** To specify the ATM virtual channel identifier (VCI), a value in the **ATM VCI** field.

The value can be from 32 to 65535.

**Step 9** Check the **UNI Shutdown** check box if you want to leave the UNI port shut during service activation (for example, when the service provider wants to deploy a service in the network but wants to activate it at a later time).

**Step 10** Check the **Use Existing PW Class** check box to enable the selection of a pseudowire class.

This attribute is unchecked by default.

Usage notes:

- If Use Existing PW Class is checked, an additional attribute, **Existing PW Class Name**, appears in the GUI. Enter the name of a pseudowire class which already exists in the device.
- If Use Existing PW Class is checked, the PW Tunnel Selection and Interface Tunnel attributes will disappear from the window. This is to prevent ISC from generating the pseudowire class.
- The Use PseudoWireClass attribute is only available if the MPLS core connectivity type was set as PSEUDOWIRE in the Service Options window (see Pseudowire Core Connectivity, page 6-3).
- Use PseudoWireClass is only applicable for IOS XR devices.

**Step 11** Check the **N-PE Pseudo-wire on SVI** check box to have ISC generate forwarding commands under SVIs (switch virtual interfaces).

By default, this check box is not checked. In this case, ISC generates forwarding commands under the service instance.

For a FlexUNI link, the attribute N-PE Pseudo-wire on SVI is dependent on the value of the attribute Configure with Bridge Domain (this is available in the service request workflow in the EVC Service Request Editor window). N-PE Pseudo-wire on SVI, if enabled, will be reflected only when Configure with Bridge Domain is set to true. Otherwise, the service request will not be created with xconnect under SVI, even if N-PE Pseudo-wire on SVI is enabled.

Usage notes:

- For an ATM link, the attribute N-PE Pseudo-wire on SVI is dependent on the value of the attribute Configure with Bridge Domain (in the EVC Service Request Editor window). N-PE Pseudo-wire on SVI, if enabled, will be reflected only when Configure with Bridge Domain is set to true. Otherwise, the service request will not be created with xconnect under SVI, even if N-PE pseudo-wire on SVI is enabled.
- ISC supports a hybrid configuration for FlexUNI/EVC service requests. In a hybrid configuration, the forwarding commands (such as xconnect) for one side of an attachment circuit can be configured under a service instance, and the xconnect configuration for the other side of the attachment circuit can be configured under a switch virtual interface (SVI).
- N-PE Pseudo-wire on SVI is applicable for all connectivity types (PSEUDOWIRE or LOCAL), but a hybrid SVI configuration is possible only for pseudowire connectivity.
- When MPLS Core Connectivity Type is set as LOCAL connectivity type, the N-PE Pseudo-wire on SVI attribute is always disabled in the policy and service request.
- For examples of these cases, see configlet examples FlexUNI/EVC (Pseudowire Core Connectivity, Bridge Domain, Pseudowire on SVI) and FlexUNI/EVC (Pseudowire Core Connectivity, no Bridge Domain, no Pseudowire on SVI).
- For additional information on the N-PE Pseudo-wire on SVI attribute, see the corresponding coverage in the FlexUNI/EVC policy chapter in the section Setting the Interface Attributes, page 3-16.

- The N-PE Pseudo-wire on SVI attribute is not supported for IOS XR devices. All the xconnect commands are configured on L2 subinterfaces/service instance.

**Step 12** Check the **PW Tunnel Selection** check box if you want to be able to manually select the Traffic Engineering (TE) tunnel for the pseudowire connecting point-to-point N-PEs.

Usage notes:

- Checking the PW Tunnel Selection check box activates the Interface Tunnel attribute field (see the next step).
- This attribute only appears if the MPLS core connectivity type is set as pseudowire in the FlexUNI/EVC policy.

**Step 13** If you checked the PW Tunnel Selection check box, enter the TE tunnel ID in the **Interface Tunnel** text field.

ISC uses the tunnel information to create and provision a pseudowire class that describes the pseudowire connection between two N-PEs. This pseudowire class can be shared by more than one pseudowire, as long as the pseudowires share the same tunnel ID and remote loopback address. During service request creation, ISC does not check the validity of the tunnel ID number. That is, ISC does not verify the existence of the tunnel.

**Step 14** Check the **AutoPick Bridge Domain/VLAN ID** check box to have ISC autopick the VLAN ID during service request creation.

If this check box is unchecked, you are prompted to specify a VLAN ID during service request creation (see the next step).

Usage notes:

- AutoPick Bridge Domain/VLAN ID consumes a global VLAN ID on the device.
- The bridge domain VLAN ID is picked from the existing ISC VLAN pool.

**Step 15** If the AutoPick Bridge Domain/VLAN ID check box is unchecked, enter an ID number in the **Bridge Domain/VLAN ID** text field.

Usage notes:

- If AutoPick Bridge Domain/VLAN ID is checked, this field is non-editable.
- When a VLAN ID is manually allocated, ISC verifies the VLAN ID to see if it lies within ISC's VLAN ID pool. If the VLAN ID is in the pool but not allocated, the VLAN ID is allocated to the service request. If the VLAN ID is in the pool and is already in use, ISC prompts you to allocate a different VLAN ID. If the VLAN ID lies outside of the ISC VLAN ID pool, ISC does not perform any verification about whether the VLAN ID allocated. The operator must ensure the VLAN ID is available.

**Step 16** Click **OK** to save the ATM-Ethernet Attributes settings and return to the FlexUNI(EVC) Service Request Editor window.

The value in the Link Attributes column now displays as "Changed," signifying that the link settings have been updated. You can edit the link attributes now or at a future time by clicking on the Changed link and modifying the settings in the Standard UNI Details window.

See Modifying the FlexUNI/EVC Service Request, page 6-23 for details on editing the link attributes.

**Step 17** To add another link click the **Add** button and set the attributes for the new link as in the previous steps in this section.

**Step 18** To delete a link, check the check box in the first column of the row for that link and click the **Delete** button.

**Step 19** If you want to set up links with L2 access nodes for this service request, see Setting Links with L2 Access Nodes, page 6-22.

**Step 20** When you have completed setting the attributes in the FlexUNI(EVC) Service Request Editor window, click the **Save** button at the bottom of the window to save the settings and create the FlexUNI/EVC service request.

If any attributes are missing or incorrectly set, ISC displays a warning in the lower left of the window. Make any corrections or updates needed (based on the information provided by ISC), and click the **Save** button.

For information on modifying a FlexUNI/EVC service request see the section Modifying the FlexUNI/EVC Service Request, page 6-23. For additional information about saving a FlexUNI/EVC service request, see Saving the FlexUNI/EVC Service Request, page 6-24.

## Setting Links with L2 Access Nodes

The Links with L2 Access Nodes section of the FlexUNI(EVC) Service Request Editor window allows you to set up links with L2 (Ethernet) access nodes. These are similar to direct connect links, except that they have L2/Ethernet access nodes beyond the N-PE (towards the CE). Therefore, NPCs are involved.

**Note** ATM links are not supported in L2 access nodes. ATM links must be set up as direct connect links. For more information, see Setting the ATM Link Attributes, page 6-18.

The steps for setting up links with L2 access nodes are similar to those covered in the section Setting Direct Connect Links, page 6-8. See that section for detailed steps on the following common operations:

- Adding and deleting links.
- Selecting the N-PE.
- Choosing the UNI interface.
- Setting the link as a FlexUNI link.
- Editing the standard and FlexUNI link attributes.

The main difference in setting up links with L2 access does is specifying the NPC details.

Perform the following steps to set the NPC details for links with L2 access nodes:

**Step 1** The first step in the process of adding a link using NPCs is selecting the U-PE/PE-AGG device, rather than the N-PE.

If only one NPC exists for the chosen interface, that NPC is autopopulated in the Circuit Details column, and you need not choose it explicitly.

If more then one NPC is available, click **Select one circuit** in the Circuit Selection column. The NPC window appears, enabling you to choose the appropriate NPC.

**Step 2** Click **OK**.

Each time you choose a PE and its interface, the NPC that was set up from this PE and interface is automatically displayed under **Circuit Selection**. This means that you do not have to further specify the PE to complete the link.

If you want to review the details of this NPC, click **Circuit Details** in the Circuit Details column. The NPC Details window appears and lists the circuit details for this NPC.

**Step 3** For details about editing link attributes, adding or deleting links, or using the FlexUNI check box, see the corresponding steps in the section Setting Direct Connect Links, page 6-8.

**Step 4** When you have completed setting the attributes in the EVC Service Request Editor window, click the **Save** button at the bottom of the window to save the settings and create the FlexUNI/EVC service request.

If any attributes are missing or incorrectly set, ISC displays a warning in the lower left of the window. Make any corrections or updates needed (based on the information provided by ISC), and click the **Save** button.

For information on modifying a FlexUNI/EVC service request see the section Modifying the FlexUNI/EVC Service Request, page 6-23. For additional information about saving a FlexUNI/EVC service request, see Saving the FlexUNI/EVC Service Request, page 6-24.

# Modifying the FlexUNI/EVC Service Request

You can modify a FlexUNI/EVC service request if you must change or modify the links or other settings of the service request.

To modify a FlexUNI/EVC service request, perform the following steps.

**Step 1** Choose **Service Inventory** > **Inventory and Connection Manager** > **Service Requests**.

The Service Requests window appears, showing service request available in ISC.

**Step 2** Check a check box for a service request.

**Step 3** Click **Edit**.

EVC Service Editor window appears.

**Step 4** Modify any of the attributes, as desired.

See the sections start with "Setting the Service Request Details" section on page 6-3 for detailed coverage of setting attributes in this window.

**Note** Once the VC ID, VPLS VPN ID, and VLAN ID have been set in a service request they cannot be modified.

**Step 5** To add a template/data file to an attachment circuit, see the section Using Templates and Data Files with a FlexUNI/EVC Service Request, page 6-24.

**Step 6** When you are finished editing the FlexUNI/EVC service request, click **Save**.

For additional information about saving a FlexUNI/EVC service request, see Saving the FlexUNI/EVC Service Request, page 6-24.

# Using Templates and Data Files with a FlexUNI/EVC Service Request

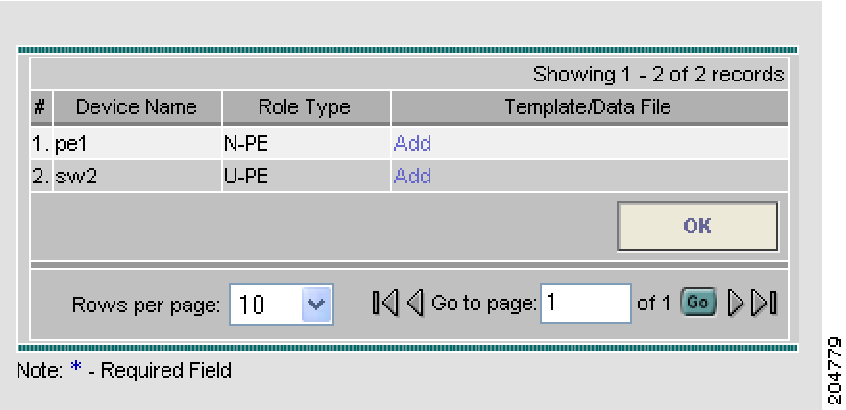
ISC does not support configuration of all the available CLI commands on a device being managed by the application. In order to configure such commands on the devices, you can use ISC Template Manager functionality. Templates can be associated at the policy level on a per-device role basis. Templates can be overridden at service request level, if the policy-level setting permits the operator to do so.

To associate templates and data files in a service request select any link in the Service Request Editor window and click the **Template** button at the bottom of the window.

**Note** If the template feature has not been enabled in the associated policy then the Template button will not be available for selection.

The SR Template Association window appears, as shown in Figure 6-8. In this window, you can associate templates at a per-device level.

***Figure 6-8 Sample Templates Association Window***



Showing 1 - 2 of 2 records

| # | Device Name | Role Type | Template/Data File |
|---|---|---|---|
| 1. | pe1 | N-PE | Add |
| 2. | sw2 | U-PE | Add |

As shown in Figure 6-8, the Template Association window lists the devices comprising the link, the device roles, and the template(s)/data file(s) associated with the devices. In this case, the template(s)/data file(s) have not yet been set up.

For further instructions on how to associate templates and data files with a service request, see Appendix B, "Working with Templates and Data Files," especially the section Using Templates with Service Requests, page B-10.

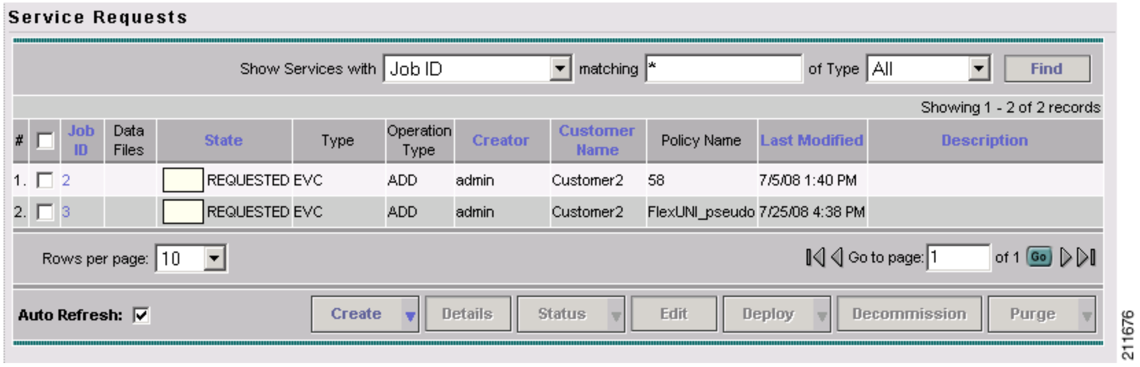
# Saving the FlexUNI/EVC Service Request

To save a FlexUNI/EVC service request, perform the following steps.

**Step 1** When you have finished setting the attributes for the FlexUNI/EVC service request, click **Save** to create the service request.

If the FlexUNI/EVC service request is successfully created, you will see the service request list window, similar to what appears in Figure 6-9.

*Figure 6-9 FlexUNI/EVC Service Request Created*



Service Requests

Show Services with: Job ID matching: * of Type: All [Find]

Showing 1 - 2 of 2 records

| # | □ | Job ID | Data Files | State | Type | Operation Type | Creator | Customer Name | Policy Name | Last Modified | Description |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1. | □ | 2 | | REQUESTED | EVC | ADD | admin | Customer2 | 58 | 7/5/08 1:40 PM | |
| 2. | □ | 3 | | REQUESTED | EVC | ADD | admin | Customer2 | FlexUNI_pseudo | 7/25/08 4:38 PM | |

Rows per page: 10

Go to page: 1 of 1

Auto Refresh: ☑

The newly created FlexUNI/EVC service request is added with the state of REQUESTED, as shown in the figure.

**Step 2** If, however, the FlexUNI service request creation failed for some reason (for example, a value chosen is out of bounds), you are warned with an error message.

In such a case, you should correct the error and save the service request again.

**Step 3** If you are ready to deploy the FlexUNI/EVC service request, see Deploying Service Requests, page 11-1.

CHAPTER 7

# Creating an L2VPN Policy

This chapter covers the basic steps to create an L2VPN policy. It contains the following sections:

## Defining an L2VPN Policy

You must define an L2VPN policy before you can provision a Cisco IP Solution Center (ISC) service. An L2VPN policy defines the common characteristics shared by the end-to-end wire attributes and Attachment Circuit (AC) attributes.

A policy is a template of most of the parameters needed to define an L2VPN service request. After you define it, an L2VPN policy can be used by all the L2VPN service requests that share a common set of characteristics. You create a new L2VPN policy whenever you create a new type of service or a service with different parameters. L2VPN policy creation is normally performed by experienced network engineers.

A policy can be shared by one or more service requests that have similar service requirements. The Editable check box gives the network operator the option of making a field editable. If the value is set to editable, the service request creator can change to other valid values for the particular policy item. If the value is *not* set to editable, the service request creator cannot change the policy item.

The four major categories of an L2VPN policy correspond to the four major services that L2VPN provides:

- Point-to-point Ethernet Relay Service (ERS). The Metro Ethernet Forum (MEF) name for this service is Ethernet Virtual Private Line (EVPL). See Layer 2 Terminology Conventions, page E-1 for more information about terms used to denote L2VPN services in this guide.
- Point-to-point Ethernet Wire Service (EWS). The MEF name for this service is Ethernet Private Line (EPL).

- Frame Relay over MPLS (FRoMPLS)
- ATM over MPLS (ATMoMPLS)

To define an L2VPN policy in ISC, perform the following steps.

---

**Step 1** Choose **Service Design > Policies**.

The Policies window appears.

**Step 2** Click **Create**.

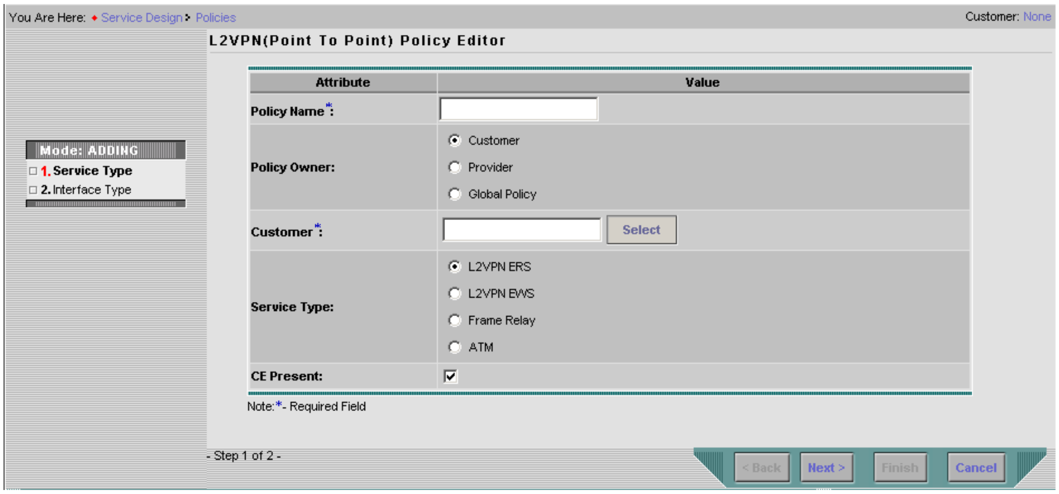
**Step 3** Choose **L2VPN (P2P) Policy**.

When you choose **L2VPN (P2P) Policy**, the L2VPN (Point to Point) Policy Creation window appears.

**Step 4** Choose **L2VPN on MPLS Core**.

The window in Figure 7-1 appears.

***Figure 7-1 Creating an L2VPN Policy***



You Are Here: ♦ Service Design ▸ Policies

Customer: None

L2VPN(Point To Point) Policy Editor

Mode: ADDING
□ 1. Service Type
□ 2. Interface Type

| Attribute | Value |
|---|---|
| Policy Name*: | |
| Policy Owner: | ● Customer<br>○ Provider<br>○ Global Policy |
| Customer*: | [Select] |
| Service Type: | ● L2VPN ERS<br>○ L2VPN EWS<br>○ Frame Relay<br>○ ATM |
| CE Present: | ☑ |

Note: *- Required Field

- Step 1 of 2 -

**Step 5** Enter a **Policy Name** for the L2VPN policy.

**Step 6** Choose the **Policy Owner** for the L2VPN policy.

There are three types of L2VPN policy ownership:

- Customer ownership
- Provider ownership
- Global ownership—Any service operator can make use of this L2VPN policy.

This ownership has relevance when the ISC Role-Based Access Control (RBAC) comes into play. For example, an L2VPN policy that is customer-owned can only be seen by operators who are allowed to work on this customer-owned policy.

Similarly, operators who are allowed to work on a provider's network can view, use, and deploy a particular provider-owned policy.

**Step 7** Click **Select** to choose the owner of the L2VPN.

(If you choose Global ownership, the Select function is not available.) The Select Customer window or the Select Provider window appears and you can choose an owner of the policy and click **Select**.

**Step 8** Choose the **Service Type** of the L2VPN policy.

There are four service types for L2VPN policies:

- L2VPN ERS (EVPL)
- L2VPN EWS (EPL)
- Frame Relay
- ATM

Subsequent sections of this chapter cover setting up the policies for each of these services.

**Step 9** Check the **CE Present** check box if you want ISC to ask the service operator who uses this L2VPN policy to provide a CE router and interface during service activation.

The default is CE present in the service.

If you do not check the **CE Present** check box, ISC asks the service operator, during service activation, only for the U-PE or the N-PE router and customer-facing interface.

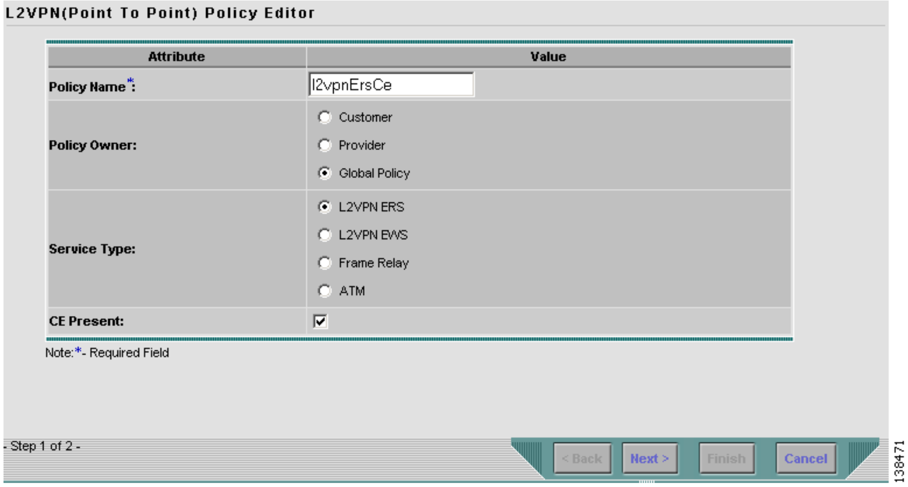
**Step 10** Click **Next**.

The next sections contain examples of setting policies for the service types, with and without a CE present.

# Defining an Ethernet ERS (EVPL) Policy with a CE

This section describes defining an Ethernet ERS (EVPL) policy with CE present. Figure 7-2 is an example of the first page of this policy.

*Figure 7-2 Ethernet ERS (EVPL) Policy with a CE*



L2VPN(Point To Point) Policy Editor

| Attribute | Value |
|---|---|
| Policy Name*: | l2vpnErsCe |
| Policy Owner: | ○ Customer<br>○ Provider<br>● Global Policy |
| Service Type: | ● L2VPN ERS<br>○ L2VPN EWS<br>○ Frame Relay<br>○ ATM |
| CE Present: | ☑ |

Note:*- Required Field

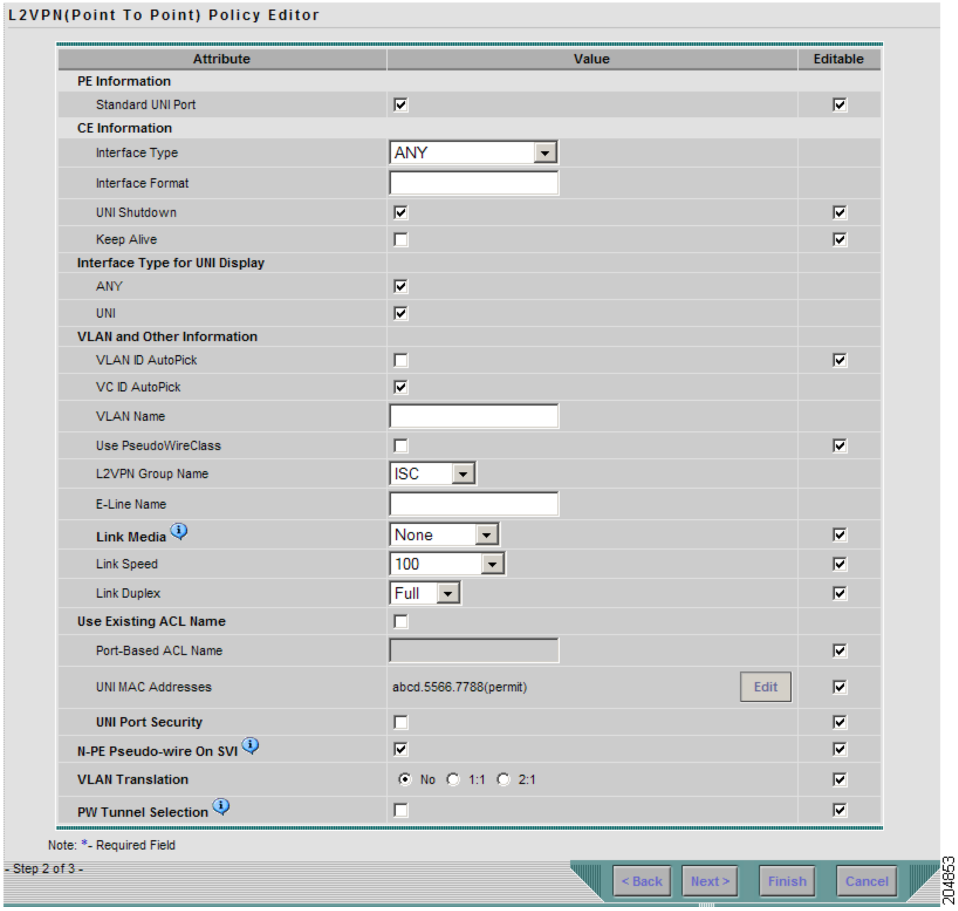
- Step 1 of 2 -

Perform the following steps.

**Step 1** Click **Next**. The window in Figure 7-3 appears.

The **Editable** check box gives you the option of making a field editable. If you check the **Editable** check box, the service operator who is using this L2VPN policy can modify the editable parameter during L2VPN service request creation.

***Figure 7-3 Ethernet ERS (EVPL) with CE Policy Attributes***



L2VPN(Point To Point) Policy Editor

| Attribute | Value | Editable |
|---|---|---|
| PE Information | | |
| Standard UNI Port | ☑ | ☑ |
| CE Information | | |
| Interface Type | ANY | |
| Interface Format | | |
| UNI Shutdown | ☑ | ☑ |
| Keep Alive | ☐ | ☑ |
| Interface Type for UNI Display | | |
| ANY | ☑ | |
| UNI | ☑ | |
| VLAN and Other Information | | |
| VLAN ID AutoPick | ☐ | ☑ |
| VC ID AutoPick | ☑ | |
| VLAN Name | | |
| Use PseudoWireClass | ☐ | ☑ |
| L2VPN Group Name | ISC | |
| E-Line Name | | |
| Link Media | None | ☑ |
| Link Speed | 100 | ☑ |
| Link Duplex | Full | ☑ |
| Use Existing ACL Name | ☐ | |
| Port-Based ACL Name | | ☑ |
| UNI MAC Addresses | abcd.5566.7788(permit) Edit | ☑ |
| UNI Port Security | ☐ | ☑ |
| N-PE Pseudo-wire On SVI | ☑ | ☑ |
| VLAN Translation | ● No ○ 1:1 ○ 2:1 | ☑ |
| PW Tunnel Selection | ☐ | ☑ |

Note: *- Required Field

- Step 2 of 3 -

**Step 2** Check the **Standard UNI Port** check box to enable port security.

This is the default. When you uncheck the check box, the port is treated as an uplink with no security features, and the window dynamically changes to eliminate items related to port security.

**Note** The **Standard UNI Port** attribute will be unavailable within service requests based on this policy if the UNI is on an N-PE device running IOS XR.

**Step 3** Choose an **Interface Type** from the drop-down list.

You can choose a particular interface on a U-PE or N-PE interface based on the service provider's POP design. The interfaces are:

- **ANY** (Any interface can be chosen.)
- **Port-Channel** (A bundle of ports that share the same characteristics—this gives the service provider the ability to aggregate bandwidth and protection.)

- **Ethernet**
- **FastEthernet**
- **GE-WAN**
- **GigabitEthernet**
- **TenGigabitEthernet**
- **TenGigE**

The value defined here functions as a filter to restrict the interface types an operator can see during L2VPN service request creation.

**Step 4** Enter an **Interface Format** as the slot number/port number for the CE interface (for example, **1/0** indicates that the interface is located at slot 1, port 0).

This is especially useful to specify here if you know that the link will always go through a particular interface's slot/port location on all or most of the network devices in the service.

**Step 5** Choose an **Encapsulation** type.

The choices are:

- **DOT1Q**
- **DEFAULT**

If **DEFAULT** is the CE encapsulation type, ISC shows another field for the UNI port type.

**Note** If the Interface Type is ANY, ISC will not ask for an **Encapsulation** type in the policy.

**Step 6** Check the **UNI Shutdown** check box if you want to leave the UNI port shut during service activation, for example, when the service provider wants to deploy a service in the network but wants to activate it at a later time.

**Step 7** Check the **Keep Alive** check box to configure keepalives on the UNI port.

By default, this check box is unchecked, which causes the command **no keepalive** to be provisioned on the UNI port. This prevents a CPE from sending keepalive packets to the U-PE, for security purposes. This attribute is editable to support modification on a per-service request basis.

**Step 8** Check the **ANY** check box to display all interface types as choices for the UNI interface (when creating service requests based on this policy).

This check box is checked by default.

**Step 9** Check the **UNI** check box to display all interfaces defined as type UNI as choices for the UNI interface (when creating service requests based on this policy).

This check box is checked by default.

**Step 10** Check the **VLAN ID AutoPick** check box if you want ISC to choose a VLAN ID.

If you do not check this check box, you will be prompted to provide the VLAN in a Provider VLAN ID field during service activation.

**Step 11** Check the **VC ID AutoPick** check box if you want ISC to choose a VC ID.

If you do not check this check box, you will be prompted to provide the VC ID in a VC ID field during service activation.

**Step 12** Enter a **VLAN NAME** (optional) to specify a name to describe the VLAN.

The name must be one token (no spaces allowed.) The limit for the VLAN name is 32 characters. The name has to be unique.

**Step 13** Check the **Use PseudoWireClass** check box to enable the selection of a pseudowire class.

This attribute is only applicable for IOS XR devices. If the check box is checked, an additional attribute, **PseudoWireClass**, appears in the GUI. Click the **Select** button of PseudoWireClass attribute to choose a pseudowire class previously created in ISC. The pseudowire class name is used for provisioning pw-class commands on IOS XR devices. See Creating and Modifying Pseudowire Classes for IOS XR Devices, page 2-10 for additional information on pseudowire class support for IOS XR devices.

**Step 14** Choose an **L2VPN Group Name** from the drop-down list.

The choices are:

- **ISC**
- **VPNSC**

This attribute is used for provisioning the L2VPN group name on IOS XR devices.

**Note** The choices in the drop-down list are derived from a configurable DCPL property. For information about how to define the L2VPN Group Name choices available in the drop-down list, see Defining L2VPN Group Names for IOS XR Devices, page 2-14.

**Step 15** Enter an **E-Line Name** to specify the point-to-point (p2p) E-line name.

This attribute is only applicable for IOS XR devices. If no value is specified for the **p2p** name, ISC generates a default name consisting of the names of the two PEs forming the pseudowire, separated by hyphens (for example, 6503-A----6503-B). If the default name is more than 32 characters, the device names are truncated.

**Step 16** Enter a **Link Media** type (optional) of None, auto-select, rj45, or sfp.

Usage notes:

- The default is None.
- When this attribute is used, a new CLI will be generated in the UNI interface to define the media type.
- The Link Media attribute is supported only for ME3400 platforms.

**Step 17** Enter a **Link Speed** (optional) of None, 10, 100, 1000, Auto, or nonegotiate.

**Step 18** Enter a **Link Duplex** (optional) of None, Full, Half, or Auto.

**Step 19** Check the **Use Existing ACL Name** check box if you want assign your own named access list to the port.

By default, this box is unchecked and ISC automatically assigns a MAC-based ACL on the customer facing UNI port, based on values you enter in **UNI MAC addresses** (below).

**Step 20** Enter a **Port-Based ACL Name** (if you checked the **Use Existing ACL Name** check box, as mentioned in the previous step).

**Note** ISC does not create this ACL automatically. The ACL must already exist on the device, or be added as part of a template, before the service request is deployed. Otherwise, deployment will fail.

**Step 21** Enter one or more Ethernet MAC addresses in **UNI MAC addresses**.

This selection is present only if you uncheck the **Use Existing ACL Name** check box. Click the **Edit** button to bring up a pop-up window in which you enter MAC addresses to be allowed or denied on the port. You can also specify a range of addresses by setting a base MAC address and a filtered MAC address.

**Step 22** Choose a **UNI Port Type**.

The choices are:

- **Access Port**
- **Trunk with Native VLAN**

**Note** Enter a UNI Port Type only if the encapsulation type is DEFAULT.

**Step 23** Check the **UNI Port Security** check box (see Figure 7-4) if you to want to provision port security-related CLIs to the UNI port by controlling the MAC addresses that are allowed to go through the interface.

a. For **Maximum Number of MAC address**, enter the number of MAC addresses allowed for port security.

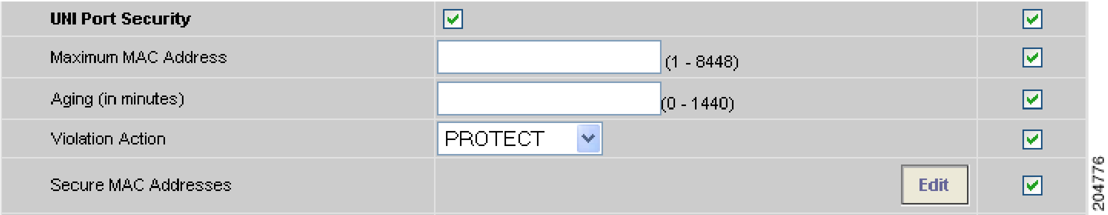
b. For **Aging,** enter the length of time the MAC address can stay on the port security table.

c. For **Violation Action**, choose what action will occur when a port security violation is detected:

- **PROTECT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value.
- **RESTRICT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value and causes the Security Violation counter to increment.
- **SHUTDOWN**—Puts the interface into the error-disabled state immediately and sends an SNMP trap notification.

d. In the **Secure MAC Addresses** field, enter one or more Ethernet MAC addresses.

*Figure 7-4 UNI Port Security*

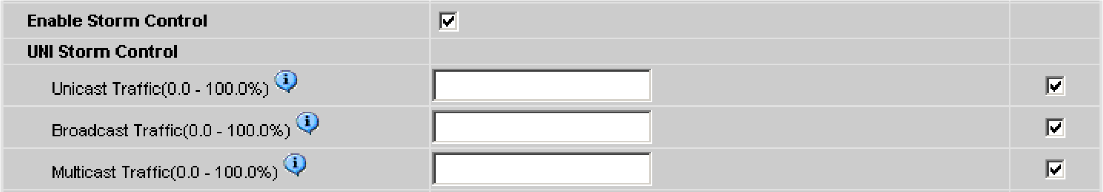


| | | |
|---|---|---|
| **UNI Port Security** | ☑ | ☑ |
| Maximum MAC Address | (1 - 8448) | ☑ |
| Aging (in minutes) | (0 - 1440) | ☑ |
| Violation Action | PROTECT | ☑ |
| Secure MAC Addresses | Edit | ☑ |

**Step 24** Check the **Enable Storm Control** check box (see Figure 7-5) to help prevent the UNI port from being disrupted by a broadcast, multicast or unicast storm.

Enter a threshold value for each type of traffic. The value, which can be specified to two significant digits, represents the percentage of the total available bandwidth of the port. If the threshold of a traffic type is reached, further traffic of that type is suppressed until the incoming traffic falls below the threshold level.

*Figure 7-5 Enable Storm Control*



| | | |
|---|---|---|
| Enable Storm Control | ☑ | |
| UNI Storm Control | | |
| Unicast Traffic(0.0 - 100.0%) | | ☑ |
| Broadcast Traffic(0.0 - 100.0%) | | ☑ |
| Multicast Traffic(0.0 - 100.0%) | | ☑ |

**Step 25** Check the **N-PE Pseudo-wire On SVI** check box to configure the pseudowire connection on the switched virtual interface of the OSM card.

This check box is checked by default. If the check box is not checked, the pseudowire will be provisioned on the subinterface of the PFC card, if it is available. This option is only available for C76xx devices.

**Note** The **N-PE Pseudo-wire on SVI** attribute will be unavailable within service requests based on this policy for devices running IOS XR.

**Step 26** Specify the type of **VLAN Translation** for this policy by clicking the appropriate radio button.

The choices are:

- **No**—No VLAN translation is performed. (This is the default.)
- **1:1**—1:1 VLAN translation.
- **2:1**—2:1 VLAN translation.

**Note** For detailed coverage of setting up VLAN translation, see Appendix C, "Setting Up VLAN Translation."

**Step 27** Check the **PW Tunnel Selection** check box if you want to be able to manually select the Traffic Engineering (TE) tunnel for the pseudowire connecting point-to-point N-PEs.

This attribute is unchecked by default

Subsequently, when you create a service request based on this policy, you must specify the TE tunnel ID in a field provided. ISC uses the tunnel information to create and provision a pseudowire class that describes the pseudowire connection between two N-PEs. This pseudowire class can be shared by more than one pseudowire, as long as the pseudowires share the same tunnel ID and remote loopback address. You are responsible to ensure that the tunnel interface and associated ID are configured. During service request creation when you specify the tunnel ID number, ISC does not check the validity of the value. That is, ISC does not verify the existence of the tunnel.

**Note** The **PW Tunnel Selection** attribute will be unavailable within service requests based on this policy for devices running IOS XR.

**Step 28** Click the **Next** button, if you want to enable template support for the policy.

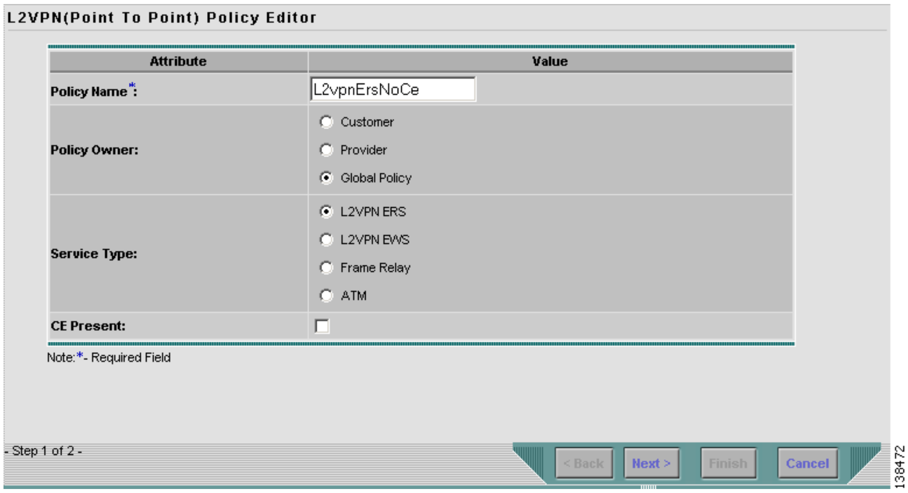
The Template Association window appears. In this window, you can enable template support and, optionally, associate templates and data files with the policy. For instructions about associating templates with policies and how to use the features in this window, see Appendix B, "Working with Templates and Data Files". When you have completed setting up templates and data files for the policy, click **Finish** in the Template Association window to close it and return to the Policy Editor window.

**Step 29** Click **Finish**.

# Defining an Ethernet ERS (EVPL) Policy without a CE

This section describes defining an Ethernet ERS (EVPL) policy without a CE present. Figure 7-6 is an example of the first page of this policy.

*Figure 7-6 Ethernet ERS (EVPL) Policy without a CE*



L2VPN(Point To Point) Policy Editor

| Attribute | Value |
|---|---|
| Policy Name*: | L2vpnErsNoCe |
| Policy Owner: | ○ Customer<br>○ Provider<br>● Global Policy |
| Service Type: | ● L2VPN ERS<br>○ L2VPN EWS<br>○ Frame Relay<br>○ ATM |
| CE Present: | □ |

Note:*- Required Field

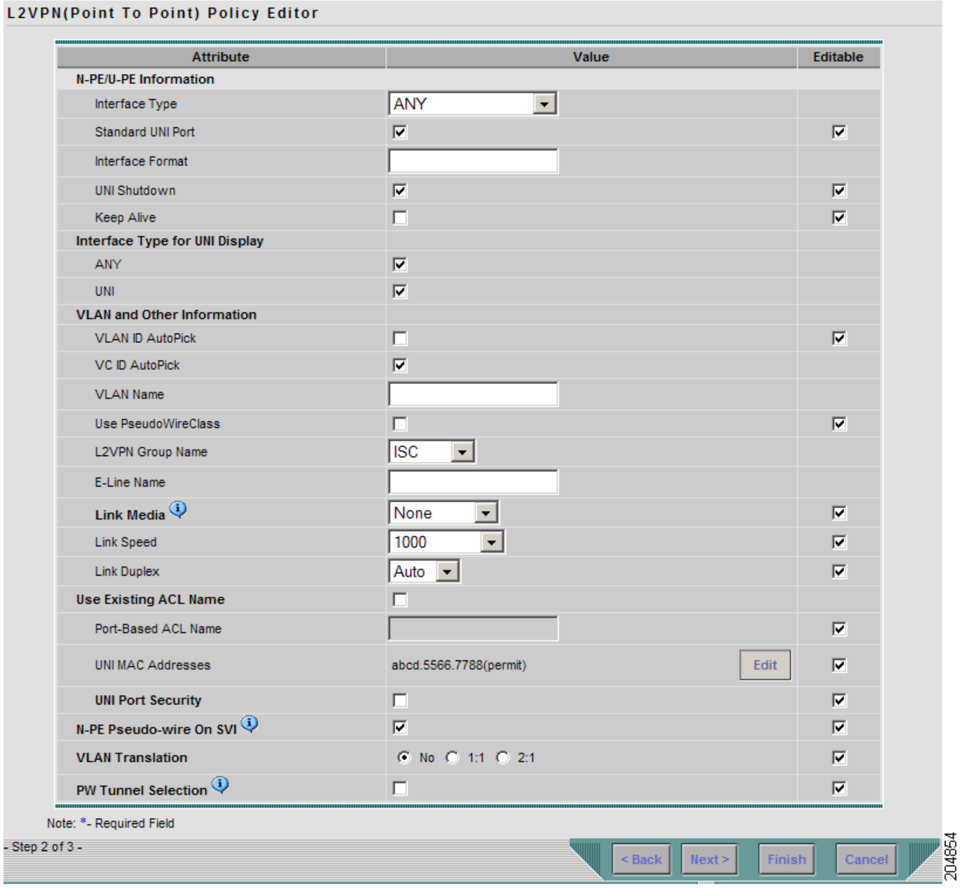
- Step 1 of 2 -

Perform the following steps.

**Step 1** Click **Next**. The window in Figure 7-7 appears.

The **Editable** check box gives you the option of making a field editable. If you check the **Editable** check box, the service operator who is using this L2VPN policy can modify the editable parameter during L2VPN service request creation.

*Figure 7-7 Ethernet ERS (EVPL) without CE Policy Attributes*



L2VPN(Point To Point) Policy Editor

| Attribute | Value | Editable |
|---|---|---|
| **N-PE/U-PE Information** | | |
| Interface Type | ANY | |
| Standard UNI Port | ☑ | ☑ |
| Interface Format | | |
| UNI Shutdown | ☑ | ☑ |
| Keep Alive | □ | ☑ |
| **Interface Type for UNI Display** | | |
| ANY | ☑ | |
| UNI | ☑ | |
| **VLAN and Other Information** | | |
| VLAN ID AutoPick | □ | ☑ |
| VC ID AutoPick | ☑ | |
| VLAN Name | | |
| Use PseudoWireClass | □ | ☑ |
| L2VPN Group Name | ISC | |
| E-Line Name | | |
| **Link Media** | None | ☑ |
| Link Speed | 1000 | ☑ |
| Link Duplex | Auto | ☑ |
| **Use Existing ACL Name** | □ | |
| Port-Based ACL Name | | ☑ |
| UNI MAC Addresses | abcd.5566.7788(permit) Edit | ☑ |
| **UNI Port Security** | □ | ☑ |
| **N-PE Pseudo-wire On SVI** | ☑ | ☑ |
| **VLAN Translation** | ● No ○ 1:1 ○ 2:1 | ☑ |
| **PW Tunnel Selection** | □ | ☑ |

Note: *- Required Field

- Step 2 of 3 -

**Step 2** Choose a N-PE/U-PE **Interface Type** from the drop-down list.

You can choose a particular interface as a CE, N-PE, or U-PE interface based on the service provider's POP design. The interfaces are:

- **ANY** (Any interface can be chosen.)
- **Port-Channel** (A bundle of ports that share the same characteristics—this gives the service provider the ability to aggregate bandwidth and protection.)
- **Ethernet**
- **FastEthernet**
- **GE-WAN**
- **GigabitEthernet**
- **TenGigabitEthernet**
- **TenGigE**

The value defined here functions as a filter to restrict the interface types an operator can see during L2VPN service request creation.

**Step 3** Check the **Standard UNI Port** check box to enable port security.

This is the default. When you uncheck the check box, the port is treated as an uplink with no security features, and the window dynamically changes to eliminate items related to port security.

**Note** The **Standard UNI Port** attribute will be unavailable within service requests based on this policy if the UNI is on an N-PE device running IOS XR.

**Step 4** Enter an **Interface Format** as the slot number/port number for the PE interface (for example, **1/0** indicates that the interface is located at slot 1, port 0).

This is especially useful to specify here if you know that the link will always go through a particular interface's slot/port location on all or most of the network devices in the service.

**Step 5** Choose an **Encapsulation** type.

The choices are:

- **DOT1Q**
- **DEFAULT**

If **DEFAULT** is the CE encapsulation type, ISC shows another field for the UNI port type.

**Note** If the Interface Type is ANY, ISC will not ask for an **Encapsulation** type in the policy.

**Step 6** Check the **UNI Shutdown** check box if you want to leave the UNI port shut during service activation, for example, when the service provider wants to deploy a service in the network but wants to activate it at a later time.

**Step 7** Check the **Keep Alive** check box to configure keepalives on the UNI port.

By default, this check box is unchecked, which causes the command **no keepalive** to be provisioned on the UNI port. This prevents a CPE from sending keepalive packets to the U-PE, for security purposes. This attribute is editable to support modification on a per-service request basis.

**Step 8** Check the **ANY** check box to display all interface types as choices for the UNI interface (when creating service requests based on this policy).

This check box is checked by default.

**Step 9** Check the **UNI** check box to display all interfaces defined as type UNI as choices for the UNI interface (when creating service requests based on this policy).

This check box is checked by default.

**Step 10** Check the **VLAN ID AutoPick** check box if you want ISC to choose a VLAN ID. If you do not check this check box, you will be prompted to provide the VLAN in a Provider VLAN ID field during service activation.

**Step 11** Check the **VC ID AutoPick** check box if you want ISC to choose a VC ID.

If you do not check this check box, you will be prompted to provide the VC ID in a VC ID field during service activation.

**Step 12** Check the **Use PseudoWireClass** check box to enable the selection of a pseudowire class.

This attribute is only applicable for IOS XR devices. If the check box is checked, an additional attribute, **PseudoWireClass**, appears in the GUI. Click the **Select** button of PseudoWireClass attribute to choose a pseudowire class previously created in ISC. The pseudowire class name is used for provisioning pw-class commands on IOS XR devices. See Creating and Modifying Pseudowire Classes for IOS XR Devices, page 2-10 for additional information on pseudowire class support for IOS XR devices.

**Step 13** Choose an **L2VPN Group Name** from the drop-down list.

The choices are:

- **ISC**
- **VPNSC**

This attribute is used for provisioning the L2VPN group name on IOS XR devices.

**Note** The choices in the drop-down list are derived from a configurable DCPL property. For information about how to define the L2VPN Group Name choices available in the drop-down list, see Defining L2VPN Group Names for IOS XR Devices, page 2-14.

**Step 14** Enter an **E-Line Name** to specify the point-to-point (p2p) E-line name.

This attribute is only applicable for IOS XR devices. If no value is specified for the p2p name, ISC generates a default name consisting of the names of the two PEs forming the pseudowire, separated by hyphens (for example, 6503-A----6503-B). If the default name is more than 32 characters, the device names are truncated.

**Step 15** Enter a **VLAN NAME** (optional) to specify a name to describe the VLAN.

The name must be one token (no spaces allowed.) The limit for the VLAN name is 32 characters. The name has to be unique.

**Step 16** Enter a **Link Media** type (optional) of None, auto-select, rj45, or sfp.

Usage notes:

- The default is None.
- When this attribute is used, a new CLI will be generated in the UNI interface to define the media type.
- The Link Media attribute is supported only for ME3400 platforms.

**Step 17** Enter a **Link Speed** (optional) of None, 10, 100, 1000, Auto, or nonegotiate.

**Step 18** Enter a **Link Duplex** (optional) of None, Full, Half, or Auto.

**Step 19** Check the **Use Existing ACL Name** check box if you want assign your own named access list to the port.

By default, this check box is unchecked and ISC automatically assigns a MAC-based ACL on the customer facing UNI port, based on values you enter in **UNI MAC addresses** (below).

**Step 20** Enter a **Port-Based ACL Name** (if you checked the **Use Existing ACL Name** check box, as mentioned in the previous step).

**Note** ISC does not create this ACL automatically. The ACL must already exist on the device, or be added as part of a template, before the service request is deployed. Otherwise, deployment will fail.

**Step 21** Enter one or more Ethernet MAC addresses in **UNI MAC addresses**.

This selection is present only if you unchecked the **Use Existing ACL Name** check box. Click the **Edit** button to bring up a pop-up window in which you enter MAC addresses to be allowed or denied on the port. You can also specify a range of addresses by setting a base MAC address and a filtered MAC address.

**Step 22** Choose a **UNI Port Type**.

The choices are:

- **Access Port**
- **Trunk with Native VLAN**

**Note** Enter a UNI Port Type only if the encapsulation type is DEFAULT.

**Step 23** Check the **UNI Port Security** check box (see Figure 7-8) if you to want to provision port security-related CLIs to the UNI port by controlling the MAC addresses that are allowed to go through the interface.

a. For **Maximum Number of MAC address**, enter the number of MAC addresses allowed for port security.

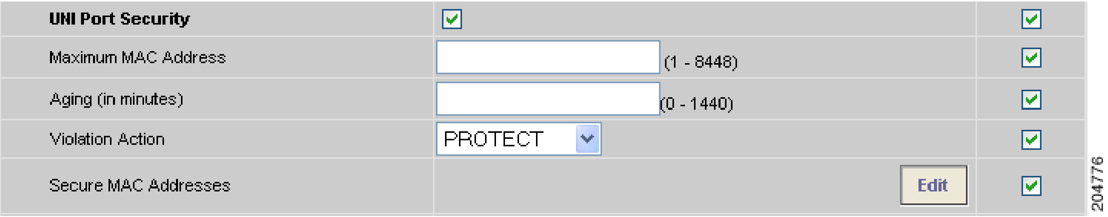
b. For **Aging,** enter the length of time the MAC address can stay on the port security table.

c. For **Violation Action**, choose what action will occur when a port security violation is detected:

- **PROTECT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value.
- **RESTRICT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value and causes the Security Violation counter to increment.
- **SHUTDOWN**—Puts the interface into the error-disabled state immediately and sends an SNMP trap notification.

d. In the **Secure MAC Addresses** field, enter one or more Ethernet MAC addresses.

***Figure 7-8 UNI Port Security***

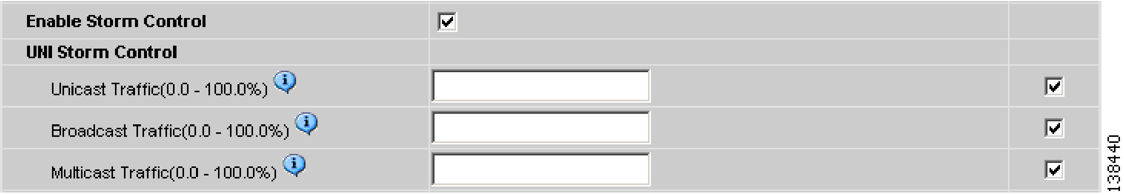


| | | |
|---|---|---|
| **UNI Port Security** | ☑ | ☑ |
| Maximum MAC Address | (1 - 8448) | ☑ |
| Aging (in minutes) | (0 - 1440) | ☑ |
| Violation Action | PROTECT | ☑ |
| Secure MAC Addresses | Edit | ☑ |

**Step 24** Check the **Enable Storm Control** check box (see Figure 7-9) to help prevent the UNI port from being disrupted by a broadcast, multicast, or unicast storm.

Enter a threshold value for each type of traffic. The value, which can be specified to two significant digits, represents the percentage of the total available bandwidth of the port. If the threshold of a traffic type is reached, further traffic of that type is suppressed until the incoming traffic falls below the threshold level.

***Figure 7-9 Enable Storm Control***



| | | |
|---|---|---|
| **Enable Storm Control** | ☑ | |
| **UNI Storm Control** | | |
| Unicast Traffic(0.0 - 100.0%) | | ☑ |
| Broadcast Traffic(0.0 - 100.0%) | | ☑ |
| Multicast Traffic(0.0 - 100.0%) | | ☑ |

**Step 25** Check the **N-PE Pseudo-wire On SVI** check box to configure the pseudowire connection on the switched virtual interface of the OSM card.

This check box is checked by default. If the check box is not checked, the pseudowire will be provisioned on the subinterface of the PFC card, if it is available. This option is only available for C76xx devices.

**Note** The **N-PE Pseudo-wire On SVI** attribute will be unavailable within service requests based on this policy for devices running IOS XR.

**Step 26** Specify the type of **VLAN Translation** for this policy by clicking the appropriate radio button.

The choices are:

- **No**—No VLAN translation is performed. (This is the default.)
- **1:1**—1:1 VLAN translation.
- **2:1**—2:1 VLAN translation.

**Note** For detailed coverage of setting up VLAN translation, see Appendix C, "Setting Up VLAN Translation."

**Step 27** Check the **PW Tunnel Selection** check box if you want to be able to manually select the Traffic Engineering (TE) tunnel for the pseudowire connecting point-to-point N-PEs.

This attribute is unchecked by default

Subsequently, when you create a service request based on this policy, you must specify the TE tunnel ID in a field provided. ISC uses the tunnel information to create and provision a pseudowire class that describes the pseudowire connection between two N-PEs. This pseudowire class can be shared by more than one pseudowire, as long as the pseudowires share the same tunnel ID and remote loopback address. You are responsible to ensure that the tunnel interface and associated ID are configured. During service request creation when you specify the tunnel ID number, ISC does not check the validity of the value. That is, ISC does not verify the existence of the tunnel.

**Note** The **PW Tunnel Selection** attribute will be unavailable within service requests based on this policy for devices running IOS XR.

**Step 28** Click the **Next** button, if you want to enable template support for the policy.

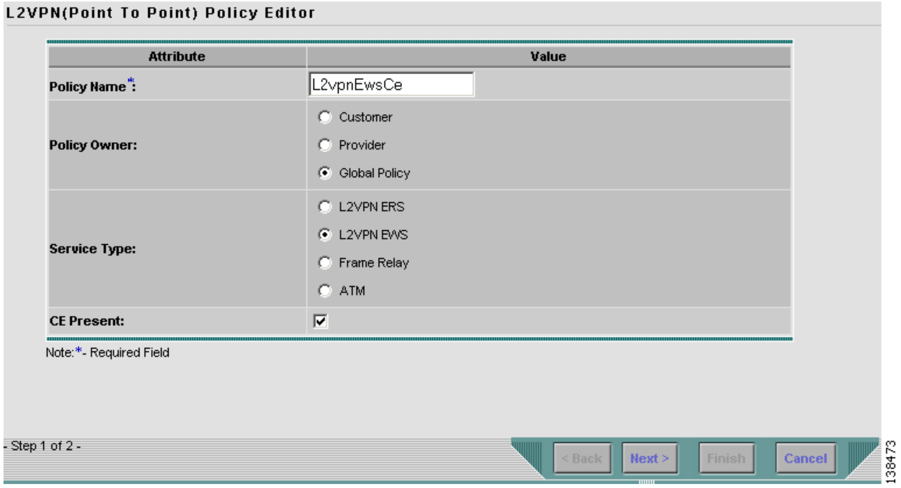
The Template Association window appears. In this window, you can enable template support and, optionally, associate templates and data files with the policy. For instructions about associating templates with policies and how to use the features in this window, see Appendix B, "Working with Templates and Data Files". When you have completed setting up templates and data files for the policy, click **Finish** in the Template Association window to close it and return to the Policy Editor window.

**Step 29** Click **Finish**.

# Defining an Ethernet EWS (EPL) Policy with a CE

This section describes defining an Ethernet EWS (EPL) policy with CE present. Figure 7-10 is an example of the first page of this policy.

*Figure 7-10 Ethernet EWS (EPL) Policy with a CE*



L2VPN(Point To Point) Policy Editor

| Attribute | Value |
|---|---|
| Policy Name*: | L2vpnEwsCe |
| Policy Owner: | ○ Customer<br>○ Provider<br>● Global Policy |
| Service Type: | ○ L2VPN ERS<br>● L2VPN EWS<br>○ Frame Relay<br>○ ATM |
| CE Present: | ☑ |

Note:*- Required Field

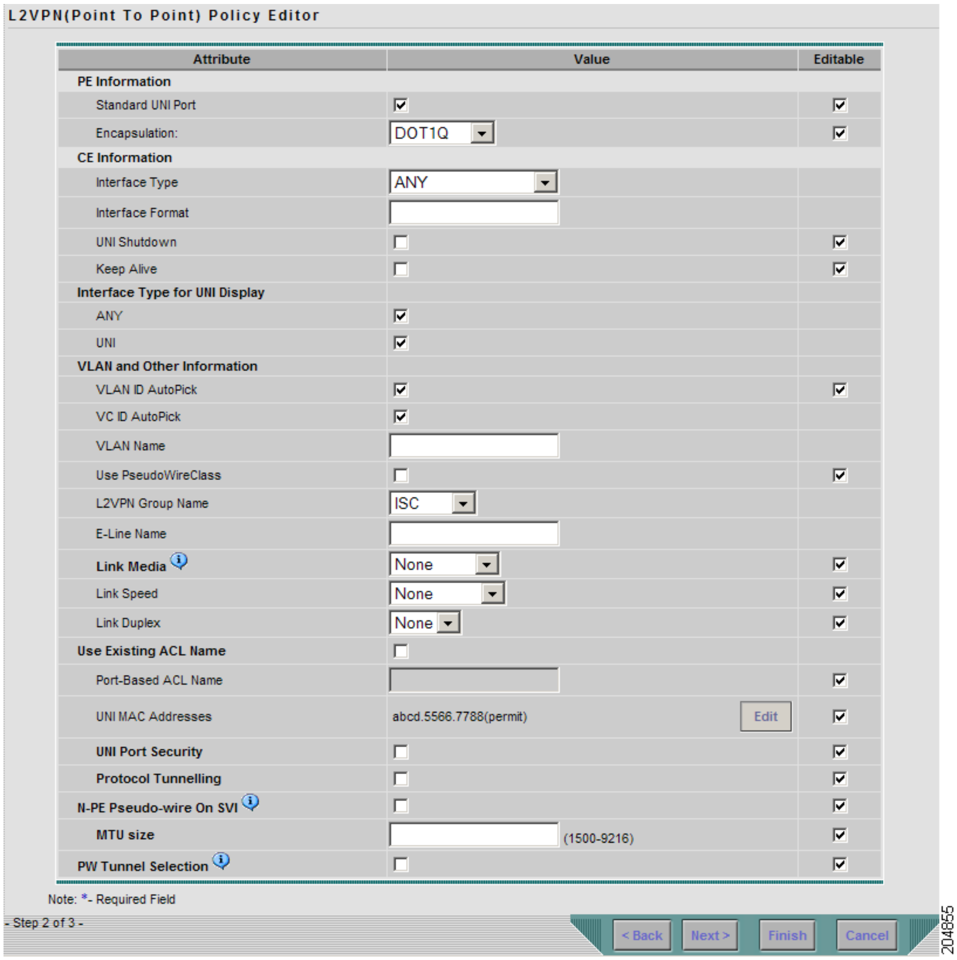
- Step 1 of 2 -

Perform the following steps.

**Step 1** Click **Next**. The window in Figure 7-11 appears.

The **Editable** check box gives you the option of making a field editable. If you check the **Editable** check box, the service operator who is using this L2VPN policy can modify the editable parameter during L2VPN service request creation.

*Figure 7-11 Ethernet EWS (EPL) with CE Policy Attributes*



L2VPN(Point To Point) Policy Editor

| Attribute | Value | Editable |
|---|---|---|
| **PE Information** | | |
| Standard UNI Port | ☑ | ☑ |
| Encapsulation: | DOT1Q | ☑ |
| **CE Information** | | |
| Interface Type | ANY | |
| Interface Format | | |
| UNI Shutdown | □ | ☑ |
| Keep Alive | □ | ☑ |
| **Interface Type for UNI Display** | | |
| ANY | ☑ | |
| UNI | ☑ | |
| **VLAN and Other Information** | | |
| VLAN ID AutoPick | ☑ | ☑ |
| VC ID AutoPick | ☑ | |
| VLAN Name | | |
| Use PseudoWireClass | □ | ☑ |
| L2VPN Group Name | ISC | |
| E-Line Name | | |
| **Link Media** | None | ☑ |
| Link Speed | None | ☑ |
| Link Duplex | None | ☑ |
| **Use Existing ACL Name** | □ | |
| Port-Based ACL Name | | ☑ |
| UNI MAC Addresses | abcd.5566.7788(permit) Edit | ☑ |
| **UNI Port Security** | □ | ☑ |
| **Protocol Tunnelling** | □ | ☑ |
| **N-PE Pseudo-wire On SVI** | □ | ☑ |
| **MTU size** | (1500-9216) | ☑ |
| **PW Tunnel Selection** | □ | ☑ |

Note: *- Required Field

- Step 2 of 3 -

**Step 2** Check the **Standard UNI Port** check box to enable port security.

This is the default. When you uncheck the check box, the port is treated as an uplink with no security features, and the window dynamically changes to eliminate items related to port security.

**Note** The **Standard UNI Port** attribute will be unavailable within service requests based on this policy if the UNI is on an N-PE device running IOS XR.

**Note** In previous releases, the only Layer 2 VPN support for EWS (EPL) was from EWS (EPL) to EWS (EPL). In ISC 4.1.2 and later, support is also from EWS (EPL) to Network to Network Interface (NNI) as a trunk port. To create this new type of service request, you need to create an EWS (EPL) "hybrid" policy by unchecking the standard UNI flag. When using the EWS (EPL) hybrid policy for service request creation, check the **Standard UNI Port flag** for the EWS (EPL) side of the connection and uncheck the standard UNI flag for the NNI side of the connection.

**Note** In the case of hybrid services, UNI on an N-PE running IOS XR is not supported.

**Step 3** Choose an **Interface Type** from the drop-down list.

You can choose a particular interface on a U-PE or N-PE interface based on the service provider's POP design. The interfaces are:

- **ANY** (Any interface can be chosen.)
- **Port-Channel** (A bundle of ports that share the same characteristics—this gives the service provider the ability to aggregate bandwidth and protection.)
- **Ethernet**
- **FastEthernet**
- **GE-WAN**
- **GigabitEthernet**
- **TenGigabitEthernet**
- **TenGigE**

The value defined here functions as a filter to restrict the interface types an operator can see during L2VPN service request creation.

**Step 4** Enter an **Interface Format** as the slot number/port number for the CE interface (for example, **1/0** indicates that the interface is located at slot 1, port 0).

This is especially useful to specify here if you know that the link will always go through a particular interface's slot/port location on all or most of the network devices in the service.

**Step 5** Choose an **Encapsulation** type.

The choices are:

- **DOT1Q**
- **DEFAULT**

If **DEFAULT** is the CE encapsulation type, ISC shows another field for the UNI port type.

**Note** If the Interface Type is ANY, ISC will not ask for an **Encapsulation** type in the policy.

**Step 6** Check the **UNI Shutdown** check box if you want to leave the UNI port shut during service activation, for example, when the service provider wants to deploy a service in the network but wants to activate it at a later time.

**Step 7** Check the **Keep Alive** check box to configure keepalives on the UNI port.

By default, this check box is unchecked, which causes the command **no keepalive** to be provisioned on the UNI port. This prevents a CPE from sending keepalive packets to the U-PE, for security purposes. This attribute is editable to support modification on a per-service request basis.

**Step 8** Check the **ANY** check box to display all interface types as choices for the UNI interface (when creating service requests based on this policy).

This check box is checked by default.

**Step 9** Check the **UNI** check box to display all interfaces defined as type UNI as choices for the UNI interface (when creating service requests based on this policy).

This check box is checked by default.

**Step 10** Check the **VLAN ID AutoPick** check box if you want ISC to choose a VLAN ID.

If you do not check this check box, you will be prompted to provide the VLAN in a Provider VLAN ID field during service activation.

**Step 11** Check the **VC ID AutoPick** check box if you want ISC to choose a VC ID.

If you do not check this check box, you will be prompted to provide the VC ID in a VC ID field during service activation.

**Step 12** Check the **Use PseudoWireClass** check box to enable the selection of a pseudowire class.

This attribute is only applicable for IOS XR devices. If the check box is checked, an additional attribute, **PseudoWireClass**, appears in the GUI. Click the **Select** button of PseudoWireClass attribute to choose a pseudowire class previously created in ISC. The pseudowire class name is used for provisioning pw-class commands on IOS XR devices. See Creating and Modifying Pseudowire Classes for IOS XR Devices, page 2-10 for additional information on pseudowire class support for IOS XR devices.

**Step 13** Choose an **L2VPN Group Name** from the drop-down list.

The choices are:

- **ISC**
- **VPNSC**

This attribute is used for provisioning the L2VPN group name on IOS XR devices.

**Note** The choices in the drop-down list are derived from a configurable DCPL property. For information about how to define the L2VPN Group Name choices available in the drop-down list, see Defining L2VPN Group Names for IOS XR Devices, page 2-14.

**Step 14** Enter an **E-Line Name** to specify the point-to-point (p2p) E-line name.

This attribute is only applicable for IOS XR devices. If no value is specified for the p2p name, ISC generates a default name consisting of the names of the two PEs forming the pseudowire, separated by hyphens (for example, 6503-A----6503-B). If the default name is more than 32 characters, the device names are truncated.

**Step 15** Enter a **VLAN NAME** (optional) to specify a name to describe the VLAN.

The name must be one token (no spaces allowed.) The limit for the VLAN name is 32 characters. The name has to be unique.

**Step 16** Check the **VLAN ID AutoPick** check box if you want ISC to choose a VLAN ID. If you do not check this check box, you will be prompted to provide the VLAN in a Provider VLAN ID field during service activation.

**Step 17** Enter a **Link Media** type (optional) of None, auto-select, rj45, or sfp.

Usage notes:

- The default is None.
- When this attribute is used, a new CLI will be generated in the UNI interface to define the media type.
- The Link Media attribute is supported only for ME3400 platforms.

**Step 18** Enter a **Link Speed** (optional) of None, 10, 100, 1000, Auto, or nonegotiate.

**Step 19** Enter a **Link Duplex** (optional) of None, Full, Half, or Auto.

**Step 20** Check the **Use Existing ACL Name** check box if you want assign your own named access list to the port. By default, this check box is not checked and ISC automatically assigns a MAC-based ACL on the customer facing UNI port, based on values you enter in **UNI MAC addresses** (below).

**Step 21** Enter a **Port-Based ACL Name** (if you checked the **Use Existing ACL Name** check box, as mentioned in the previous step).

**Note** ISC does not create this ACL automatically. The ACL must already exist on the device, or be added as part of a template, before the service request is deployed. Otherwise, deployment will fail.

**Step 22** Enter one or more Ethernet MAC addresses in **UNI MAC addresses**.

This selection is present only if you uncheck the **Use Existing ACL Name** check box. Click the **Edit** button to bring up a pop-up window in which you enter MAC addresses to be allowed or denied on the port. You can also specify a range of addresses by setting a base MAC address and a filtered MAC address.

**Step 23** Check the **UNI Port Security** check box (see Figure 7-12) if you to want to provision port security-related CLIs to the UNI port by controlling the MAC addresses that are allowed to go through the interface.

a. For **Maximum Number of MAC address**, enter the number of MAC addresses allowed for port security.

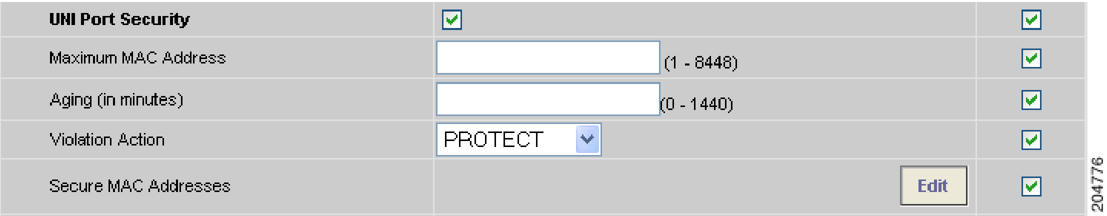
b. For **Aging,** enter the length of time the MAC address can stay on the port security table.

c. For **Violation Action**, choose what action will occur when a port security violation is detected:

- **PROTECT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value.
- **RESTRICT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value and causes the Security Violation counter to increment.
- **SHUTDOWN**—Puts the interface into the error-disabled state immediately and sends an SNMP trap notification.

d. In the **Secure MAC Addresses** field, enter one or more Ethernet MAC addresses.

*Figure 7-12 UNI Port Security*

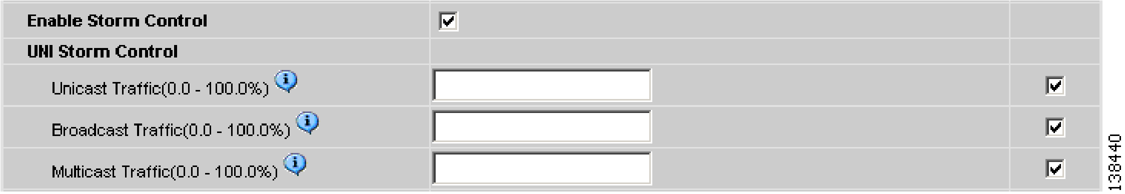


| | | |
|---|---|---|
| **UNI Port Security** | ☑ | ☑ |
| Maximum MAC Address | (1 - 8448) | ☑ |
| Aging (in minutes) | (0 - 1440) | ☑ |
| Violation Action | PROTECT | ☑ |
| Secure MAC Addresses | Edit | ☑ |

**Step 24** Check the **Enable Storm Control** check box (see Figure 7-13) to help prevent the UNI port from being disrupted by a broadcast, multicast, or unicast storm. Enter a threshold value for each type of traffic.

The value, which can be specified to two significant digits, represents the percentage of the total available bandwidth of the port. If the threshold of a traffic type is reached, further traffic of that type is suppressed until the incoming traffic falls below the threshold level.

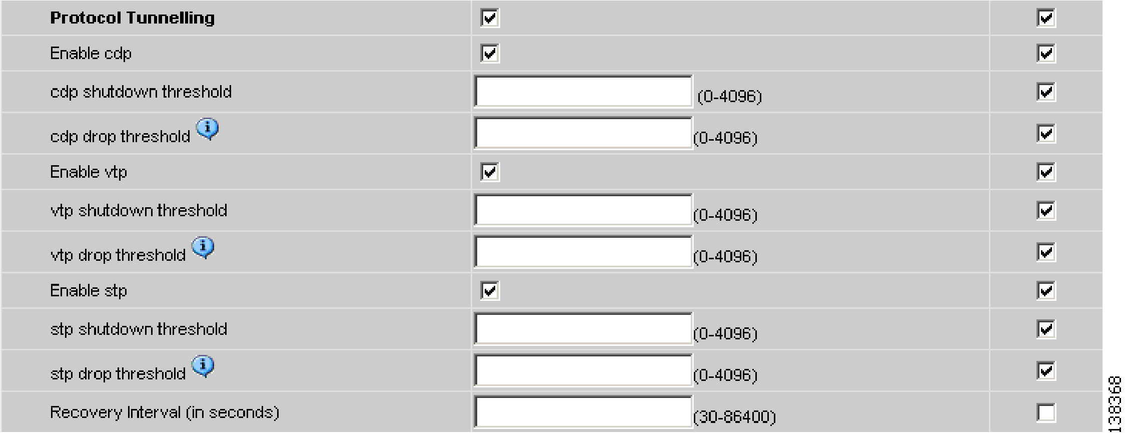
*Figure 7-13 Enable Storm Control*



|  |  |  |
|---|---|---|
| **Enable Storm Control** | ☑ |  |
| **UNI Storm Control** |  |  |
| Unicast Traffic(0.0 - 100.0%) |  | ☑ |
| Broadcast Traffic(0.0 - 100.0%) |  | ☑ |
| Multicast Traffic(0.0 - 100.0%) |  | ☑ |

**Step 25** Check the **Protocol Tunnelling** check box (see Figure 7-14) if you want to define the Layer 2 Bridge Protocol Data Unit (BPDU) frames that can be tunneled over the core to the other end.

*Figure 7-14 Protocol Tunnelling*



|  |  |  |
|---|---|---|
| **Protocol Tunnelling** | ☑ | ☑ |
| Enable cdp | ☑ | ☑ |
| cdp shutdown threshold | (0-4096) | ☑ |
| cdp drop threshold | (0-4096) | ☑ |
| Enable vtp | ☑ | ☑ |
| vtp shutdown threshold | (0-4096) | ☑ |
| vtp drop threshold | (0-4096) | ☑ |
| Enable stp | ☑ | ☑ |
| stp shutdown threshold | (0-4096) | ☑ |
| stp drop threshold | (0-4096) | ☑ |
| Recovery Interval (in seconds) | (30-86400) | □ |

For each protocol that you choose, enter the shutdown threshold and drop threshold for that protocol:

a. **Enable cdp**—Enable Layer 2 tunnelling on Cisco Discover Protocol (CDP).

b. **cdp shutdown threshold—**Enter the number of packets per second to be received before the interface is shut down.

c. **cdp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping CDP packets.

d. **Enable vtp**—Enable Layer 2 tunnelling on VLAN Trunk Protocol (VTP).

e. **vtp shutdown threshold**—Enter the number of packets per second to be received before the interface is shut down.

f. **vtp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping VTP packets.

g. **Enable stp—**Enable Layer 2 tunnelling on Spanning Tree Protocol (STP).

h. **stp shutdown threshold—**Enter the number of packets per second to be received before the interface is shut down.

i. **stp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping STP packets.

j. **Recovery Interval**—Enter the amount of time, in seconds, to wait before recovering a UNI port.

**Step 26** Check the **N-PE Pseudo-wire On SVI** check box to configure the pseudowire connection on the switched virtual interface of the OSM card.

This check box is checked by default. If the check box is not checked, the pseudowire will be provisioned on the subinterface of the PFC card, if it is available. This option is only available for C76xx devices.

**Note** The **N-PE Pseudo-wire On SVI** attribute will be unavailable within service requests based on this policy for devices running IOS XR.

**Step 27** Enter the **MTU Size** in bytes.

The maximum transmission unit (MTU) size is configurable and optional. The default size is 9216, and the range is 1500 to 9216. ISC does not perform an integrity check for this customized value. If a service request goes to the Failed Deploy state because this size is not accepted, you must adjust the size until the Service Request is deployed.

In ISC 6.0, different platforms support different ranges.

- For the 3750 and 3550 platforms, the MTU range is 1500-1546.
- For the 7600 ethernet port, the MTU size is always 9216. Even with the same platform and same IOS release, different line cards support the MTU differently. For example, older line cards only take an MTU size of 9216 and newer cards support 1500-9216. However, ISC 6.0 uses 9216 in both cases.
- For the 7600 SVI (interface VLAN), the MTU size is 1500-9216.

**Step 28** Check the **PW Tunnel Selection** check box if you want to be able to manually select the Traffic Engineering (TE) tunnel for the pseudowire connecting point-to-point N-PEs.

This attribute is unchecked by default

Subsequently, when you create a service request based on this policy, you must specify the TE tunnel ID in a field provided. ISC uses the tunnel information to create and provision a pseudowire class that describes the pseudowire connection between two N-PEs. This pseudowire class can be shared by more than one pseudowire, as long as the pseudowires share the same tunnel ID and remote loopback address. You are responsible to ensure that the tunnel interface and associated ID are configured. During service request creation when you specify the tunnel ID number, ISC does not check the validity of the value. That is, ISC does not verify the existence of the tunnel.

**Note** The **PW Tunnel Selection** attribute will be unavailable within service requests based on this policy for devices running IOS XR.

**Step 29** Click the **Next** button, if you want to enable template support for the policy.

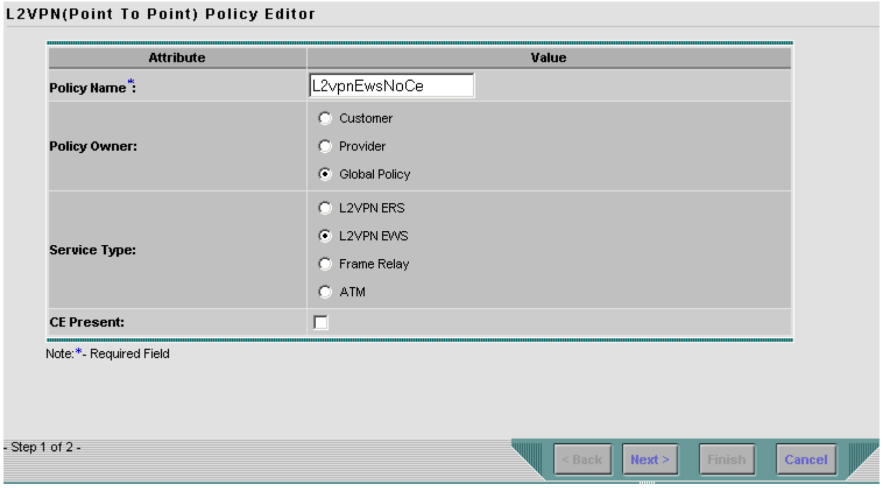
The Template Association window appears. In this window, you can enable template support and, optionally, associate templates and data files with the policy. For instructions about associating templates with policies and how to use the features in this window, see Appendix B, “Working with Templates and Data Files”. When you have completed setting up templates and data files for the policy, click **Finish** in the Template Association window to close it and return to the Policy Editor window.

**Step 30** Click **Finish**.

# Defining an Ethernet EWS (EPL) Policy without a CE

This section describes how to define an Ethernet EWS (EPL) policy without a CE present. Figure 7-15 is an example of the first page of this policy.

*Figure 7-15 Ethernet EWS (EPL) Policy without a CE*



L2VPN(Point To Point) Policy Editor

| Attribute | Value |
|---|---|
| Policy Name*: | L2vpnEwsNoCe |
| Policy Owner: | ○ Customer<br>○ Provider<br>● Global Policy |
| Service Type: | ○ L2VPN ERS<br>● L2VPN EWS<br>○ Frame Relay<br>○ ATM |
| CE Present: | □ |

Note:*- Required Field

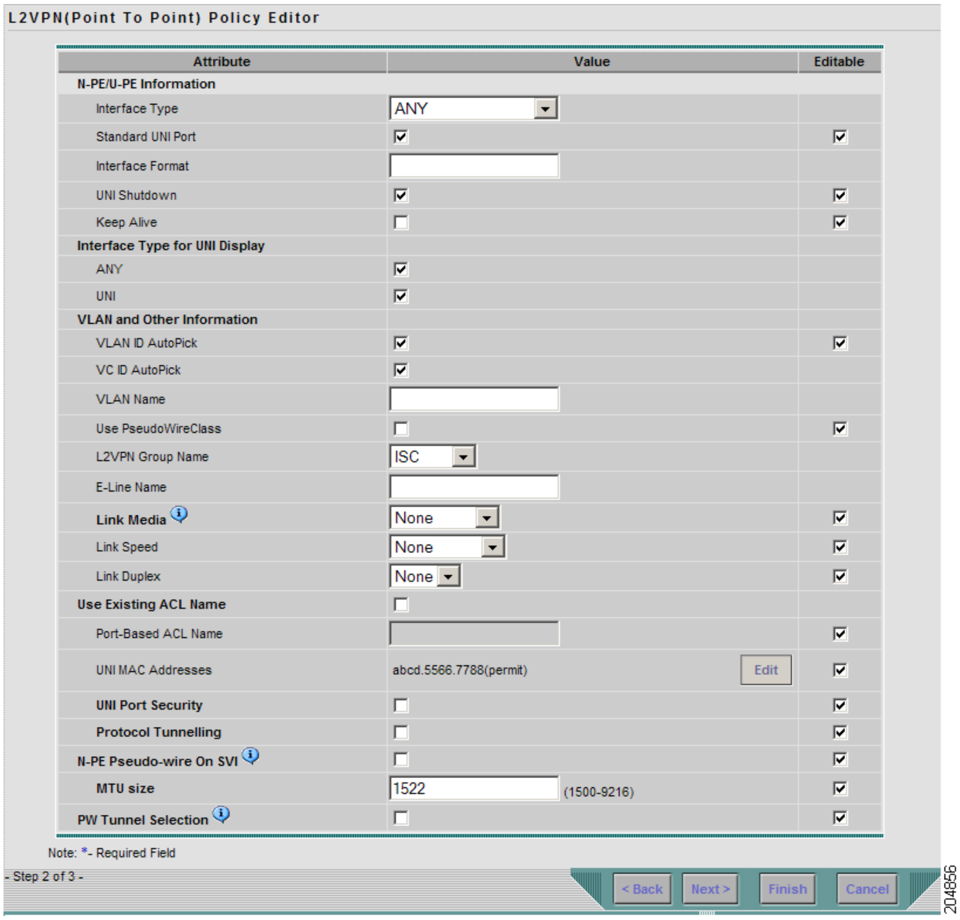
- Step 1 of 2 -

Perform the following steps.

**Step 1** Click **Next**. The window in Figure 7-16 appears.

The **Editable** check box gives you the option of making a field editable. If you check the **Editable** check box, the service operator who is using this L2VPN policy can modify the editable parameter during L2VPN service request creation.

*Figure 7-16 Ethernet EWS (EPL) without CE Policy Attributes*



L2VPN(Point To Point) Policy Editor

| Attribute | Value | Editable |
|---|---|---|
| **N-PE/U-PE Information** | | |
| Interface Type | ANY | |
| Standard UNI Port | ☑ | ☑ |
| Interface Format | | |
| UNI Shutdown | ☑ | ☑ |
| Keep Alive | □ | ☑ |
| **Interface Type for UNI Display** | | |
| ANY | ☑ | |
| UNI | ☑ | |
| **VLAN and Other Information** | | |
| VLAN ID AutoPick | ☑ | ☑ |
| VC ID AutoPick | ☑ | |
| VLAN Name | | |
| Use PseudoWireClass | □ | ☑ |
| L2VPN Group Name | ISC | |
| E-Line Name | | |
| **Link Media** | None | ☑ |
| Link Speed | None | ☑ |
| Link Duplex | None | ☑ |
| **Use Existing ACL Name** | □ | |
| Port-Based ACL Name | | ☑ |
| UNI MAC Addresses | abcd.5566.7788(permit) Edit | ☑ |
| **UNI Port Security** | □ | ☑ |
| **Protocol Tunnelling** | □ | ☑ |
| **N-PE Pseudo-wire On SVI** | □ | ☑ |
| **MTU size** | 1522 (1500-9216) | ☑ |
| **PW Tunnel Selection** | □ | ☑ |

Note: *- Required Field

- Step 2 of 3 -

**Step 2** Choose a N-PE/U-PE **Interface Type** from the drop-down list.

You can choose a particular interface as a CE, N-PE, or U-PE interface based on the service provider's POP design. The interfaces are:

- **ANY** (Any interface can be chosen.)
- **Port-Channel** (A bundle of ports that share the same characteristics—this gives the service provider the ability to aggregate bandwidth and protection.)
- **Ethernet**
- **FastEthernet**
- **GE-WAN**
- **GigabitEthernet**
- **TenGigabitEthernet**
- **TenGigE**

The value defined here functions as a filter to restrict the interface types an operator can see during L2VPN service request creation.

**Step 3** Check the **Standard UNI Port** check box to enable port security.

This is the default. When you uncheck the check box, the port is treated as an uplink with no security features, and the window dynamically changes to eliminate items related to port security.

**Note** The **Standard UNI Port** attribute will be unavailable within service requests based on this policy if the UNI is on an N-PE device running IOS XR.

**Note** In previous releases, the only Layer 2 VPN support for EWS (EPL) was from EWS (EPL) to EWS (EPL). In ISC 4.1.2 and later, support is also from EWS (EPL) to Network to Network Interface (NNI) as a trunk port. To create this new type of service request, you need to create an EWS (EPL) "hybrid" policy by unchecking the standard UNI flag. When using the EWS (EPL) hybrid policy for service request creation, check the **Standard UNI Port flag** for the EWS (EPL) side of the connection and uncheck the standard UNI flag for the NNI side of the connection.

**Note** In the case of hybrid services, UNI on an N-PE running IOS XR is not supported.

**Step 4** Enter an **Interface Format** as the slot number/port number for the PE interface (for example, **1/0** indicates that the interface is located at slot 1, port 0).

This is especially useful to specify here if you know that the link will always go through a particular interface's slot/port location on all or most of the network devices in the service.

**Step 5** Choose an **Encapsulation** type.

The choices are:

- **DOT1Q**
- **DEFAULT**

If **DEFAULT** is the CE encapsulation type, ISC shows another field for the UNI port type.

**Note** If the Interface Type is ANY, ISC will not ask for an **Encapsulation** type in the policy.

**Step 6** Check the **UNI Shutdown** check box if you want to leave the UNI port shut during service activation, for example, when the service provider wants to deploy a service in the network but wants to activate it at a later time.

**Step 7** Check the **Keep Alive** check box to configure keepalives on the UNI port.

By default, this check box is unchecked, which causes the command **no keepalive** to be provisioned on the UNI port. This prevents a CPE from sending keepalive packets to the U-PE, for security purposes. This attribute is editable to support modification on a per-service request basis.

**Step 8** Check the **ANY** check box to display all interface types as choices for the UNI interface (when creating service requests based on this policy).

This check box is not checked by default.

**Step 9** Check the **UNI** check box to display all interfaces defined as type UNI as choices for the UNI interface (when creating service requests based on this policy).

This check box is not checked by default.

**Step 10** Check the **VLAN ID AutoPick** check box if you want ISC to choose a VLAN ID.

If you do not check this check box, you will be prompted to provide the VLAN in a Provider VLAN ID field during service activation.

**Step 11** Check the **VC ID AutoPick** check box if you want ISC to choose a VC ID.

If you do not check this check box, you will be prompted to provide the VC ID in a VC ID field during service activation.

**Step 12** Check the **Use PseudoWireClass** check box to enable the selection of a pseudowire class.

This attribute is only applicable for IOS XR devices. If the check box is checked, an additional attribute, **PseudoWireClass**, appears in the GUI. Click the **Select** button of PseudoWireClass attribute to choose a pseudowire class previously created in ISC. The pseudowire class name is used for provisioning pw-class commands on IOS XR devices. See Creating and Modifying Pseudowire Classes for IOS XR Devices, page 2-10 for additional information on pseudowire class support for IOS XR devices.

**Step 13** Choose an **L2VPN Group Name** from the drop-down list.

The choices are:

- **ISC**
- **VPNSC**

This attribute is used for provisioning the L2VPN group name on IOS XR devices.

**Note** The choices in the drop-down list are derived from a configurable DCPL property. For information about how to define the L2VPN Group Name choices available in the drop-down list, see Defining L2VPN Group Names for IOS XR Devices, page 2-14.

**Step 14** Enter an **E-Line Name** to specify the point-to-point (p2p) E-line name.

This attribute is only applicable for IOS XR devices. If no value is specified for the p2p name, ISC generates a default name consisting of the names of the two PEs forming the pseudowire, separated by hyphens (for example, 6503-A----6503-B). If the default name is more than 32 characters, the device names are truncated.

**Step 15** Enter a **VLAN NAME** (optional) to specify a name to describe the VLAN.

The name must be one token (no spaces allowed.) The limit for the VLAN name is 32 characters. The name has to be unique.

**Step 16** Enter a **Link Media** type (optional) of None, auto-select, rj45, or sfp.

Usage notes:

- The default is None.
- When this attribute is used, a new CLI will be generated in the UNI interface to define the media type.
- The Link Media attribute is supported only for ME3400 platforms.

**Step 17** Enter a **Link Speed** (optional) of None, 10, 100, 1000, Auto, or nonegotiate.

**Step 18** Enter a **Link Duplex** (optional) of None, Full, Half, or Auto.

**Step 19** Check the **Use Existing ACL Name** check box if you want assign your own named access list to the port.

By default, this check box is not checked and ISC automatically assigns a MAC-based ACL on the customer facing UNI port, based on values you enter in **UNI MAC addresses** (below).

**Step 20** Enter a **Port-Based ACL Name** (if you checked the **Use Existing ACL Name** check box, as mentioned in the previous step).

**Note** ISC does not create this ACL automatically. The ACL must already exist on the device, or be added as part of a template, before the service request is deployed. Otherwise, deployment will fail.

**Step 21** Enter one or more Ethernet MAC addresses in **UNI MAC addresses**.

This selection is present only if you uncheck the **Use Existing ACL Name** check box. Click the **Edit** button to bring up a pop-up window in which you enter MAC addresses to be allowed or denied on the port. You can also specify a range of addresses by setting a base MAC address and a filtered MAC address.

**Step 22** Check the **UNI Port Security** check box (see Figure 7-4) if you to want to provision port security-related CLIs to the UNI port by controlling the MAC addresses that are allowed to go through the interface.

**a.** For **Maximum Number of MAC address**, enter the number of MAC addresses allowed for port security.

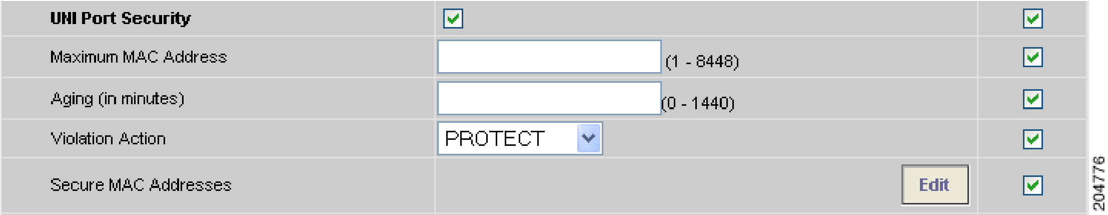
**b.** For **Aging,** enter the length of time the MAC address can stay on the port security table.

**c.** For **Violation Action**, choose what action will occur when a port security violation is detected:

- **PROTECT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value.
- **RESTRICT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value and causes the Security Violation counter to increment.
- **SHUTDOWN**—Puts the interface into the error-disabled state immediately and sends an SNMP trap notification.

**d.** In the **Secure MAC Addresses** field, enter one or more Ethernet MAC addresses.

*Figure 7-17 UNI Port Security*

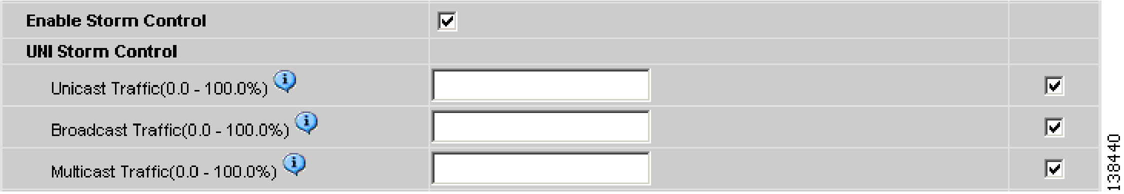


| | | |
|---|---|---|
| **UNI Port Security** | ☑ | ☑ |
| Maximum MAC Address | (1 - 8448) | ☑ |
| Aging (in minutes) | (0 - 1440) | ☑ |
| Violation Action | PROTECT | ☑ |
| Secure MAC Addresses | Edit | ☑ |

**Step 23** Check the **Enable Storm Control** check box (see Figure 7-18) to help prevent the UNI port from being disrupted by a broadcast, multicast or unicast storm.

Enter a threshold value for each type of traffic. The value, which can be specified to two significant digits, represents the percentage of the total available bandwidth of the port. If the threshold of a traffic type is reached, further traffic of that type is suppressed until the incoming traffic falls below the threshold level.

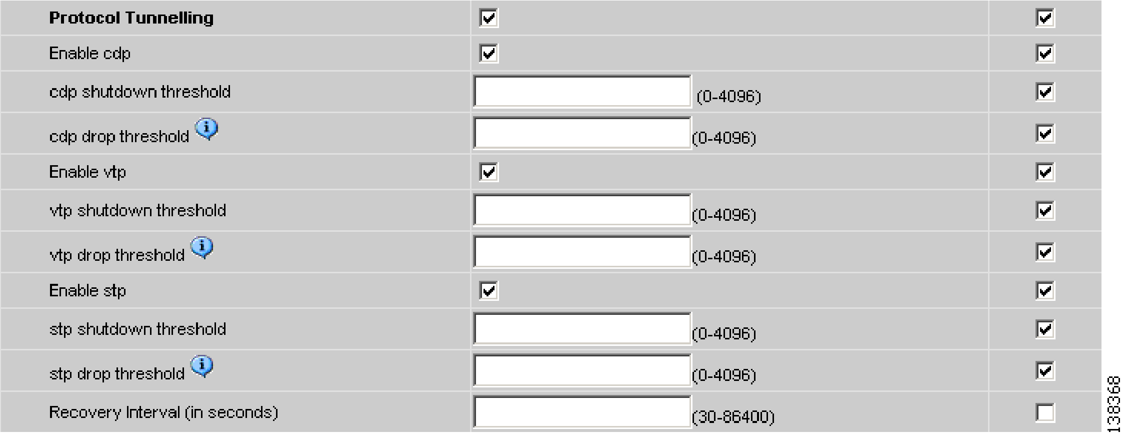
*Figure 7-18 Enable Storm Control*



| | | |
|---|---|---|
| **Enable Storm Control** | ☑ | |
| **UNI Storm Control** | | |
| Unicast Traffic(0.0 - 100.0%) ⓘ | | ☑ |
| Broadcast Traffic(0.0 - 100.0%) ⓘ | | ☑ |
| Multicast Traffic(0.0 - 100.0%) ⓘ | | ☑ |

**Step 24** Check the **Protocol Tunnelling** check box (see Figure 7-14) if you want to define the Layer 2 Bridge Protocol Data Unit (BPDU) frames that can be tunneled over the core to the other end.

*Figure 7-19 Protocol Tunnelling*



| | | |
|---|---|---|
| **Protocol Tunnelling** | ☑ | ☑ |
| Enable cdp | ☑ | ☑ |
| cdp shutdown threshold | (0-4096) | ☑ |
| cdp drop threshold ⓘ | (0-4096) | ☑ |
| Enable vtp | ☑ | ☑ |
| vtp shutdown threshold | (0-4096) | ☑ |
| vtp drop threshold ⓘ | (0-4096) | ☑ |
| Enable stp | ☑ | ☑ |
| stp shutdown threshold | (0-4096) | ☑ |
| stp drop threshold ⓘ | (0-4096) | ☑ |
| Recovery Interval (in seconds) | (30-86400) | ☐ |

For each protocol that you check, enter the shutdown threshold and drop threshold for that protocol:

a. **Enable cdp**—Enable Layer 2 tunnelling on Cisco Discover Protocol (CDP).

b. **cdp shutdown threshold—**Enter the number of packets per second to be received before the interface is shut down.

c. **cdp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping CDP packets.

d. **Enable vtp**—Enable Layer 2 tunnelling on VLAN Trunk Protocol (VTP).

e. **vtp shutdown threshold**—Enter the number of packets per second to be received before the interface is shut down.

f. **vtp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping VTP packets.

g. **Enable stp—**Enable Layer 2 tunnelling on Spanning Tree Protocol (STP).

h. **stp shutdown threshold—**Enter the number of packets per second to be received before the interface is shut down.

i. **stp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping STP packets.

j. **Recovery Interval**—Enter the amount of time, in seconds, to wait before recovering a UNI port.

**Step 25** Check the **N-PE Pseudo-wire On SVI** check box to configure the pseudowire connection on the switched virtual interface of the OSM card.

This check box is checked by default. If the check box is not checked, the pseudowire will be provisioned on the subinterface of the PFC card, if it is available. This option is only available for C76xx devices.

**Note** The **N-PE Pseudo-wire On SVI** attribute will be unavailable within service requests based on this policy for devices running IOS XR.

**Step 26** Enter the **MTU Size** in bytes.

The maximum transmission unit (MTU) size is configurable and optional. The default size is 9216, and the range is 1500 to 9216. ISC does not perform an integrity check for this customized value. If a service request goes to the Failed Deploy state because this size is not accepted, you must adjust the size until the Service Request is deployed.

In ISC 6.0, different platforms support different ranges.

- For the 3750 and 3550 platforms, the MTU range is 1500-1546.
- For the 7600 ethernet port, the MTU size is always 9216. Even with the same platform and same IOS release, different line cards support the MTU differently. For example, older line cards only take an MTU size of 9216 and newer cards support 1500-9216. However, ISC 6.0 uses 9216 in both cases.
- For the 7600 SVI (interface VLAN), the MTU size is 1500-9216.

**Step 27** Check the **PW Tunnel Selection** check box if you want to be able to manually select the Traffic Engineering (TE) tunnel for the pseudowire connecting point-to-point N-PEs.

This attribute is unchecked by default

Subsequently, when you create a service request based on this policy, you must specify the TE tunnel ID in a field provided. ISC uses the tunnel information to create and provision a pseudowire class that describes the pseudowire connection between two N-PEs. This pseudowire class can be shared by more than one pseudowire, as long as the pseudowires share the same tunnel ID and remote loopback address. You are responsible to ensure that the tunnel interface and associated ID are configured. During service request creation when you specify the tunnel ID number, ISC does not check the validity of the value. That is, ISC does not verify the existence of the tunnel.

**Note** The **PW Tunnel Selection** attribute will be unavailable within service requests based on this policy for devices running IOS XR.

**Step 28** Click the **Next** button, if you want to enable template support for the policy.

The Template Association window appears. In this window, you can enable template support and, optionally, associate templates and data files with the policy. For instructions about associating templates with policies and how to use the features in this window, see Appendix B, “Working with Templates and Data Files”. When you have completed setting up templates and data files for the policy, click **Finish** in the Template Association window to close it and return to the Policy Editor window.

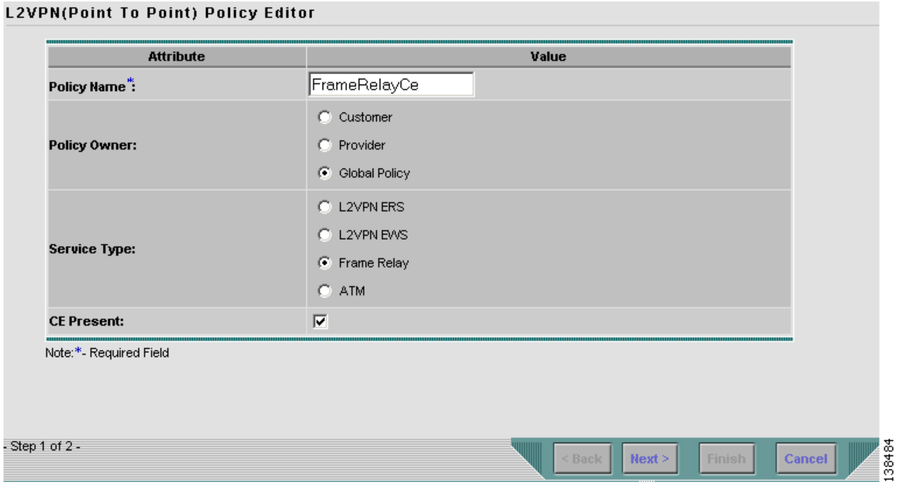
**Step 29** Click **Finish**.

# Defining a Frame Relay Policy with a CE

This section describes how to define a Frame Relay policy with CE present. Figure 7-20 is an example of the first page of this policy.

**Note** Frame Relay policies are not supported for devices running IOS XR.

*Figure 7-20 Frame Relay Policy with a CE*



L2VPN(Point To Point) Policy Editor

| Attribute | Value |
|---|---|
| Policy Name*: | FrameRelayCe |
| Policy Owner: | ○ Customer<br>○ Provider<br>● Global Policy |
| Service Type: | ○ L2VPN ERS<br>○ L2VPN EWS<br>● Frame Relay<br>○ ATM |
| CE Present: | ☑ |

Note:*- Required Field

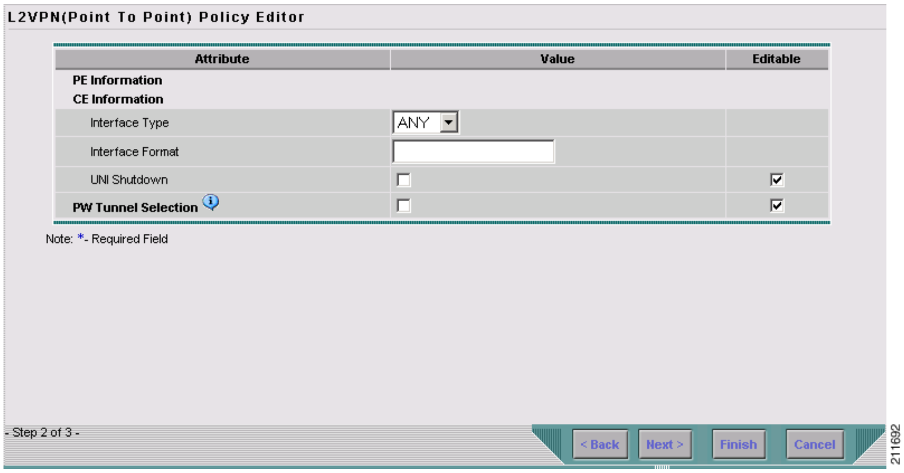
- Step 1 of 2 -

Perform the following steps.

**Step 1** Click **Next**. The window in Figure 7-21 appears.

The **Editable** check box gives you the option of making a field editable. If you check the **Editable** check box, the service operator who is using this L2VPN policy can modify the editable parameter during L2VPN service request creation.

*Figure 7-21 Frame Relay with CE Policy Attributes*



L2VPN(Point To Point) Policy Editor

| Attribute | Value | Editable |
|---|---|---|
| PE Information | | |
| CE Information | | |
| Interface Type | ANY | |
| Interface Format | | |
| UNI Shutdown | □ | ☑ |
| PW Tunnel Selection | □ | ☑ |

Note: *- Required Field

- Step 2 of 3 -

**Step 2** Choose the **Interface Type** for the **CE** from the drop-down list.

The choices are:

- **ANY**

- **Serial**
- **MFR**
- **POS**
- **Hssi**
- **BRI**

**Step 3** Enter an **Interface Format** as the slot number/port number for the CE interface (for example, **1/0** indicates that the interface is located at slot 1, port 0).

This is especially useful to specify here if you know that the link will always go through a particular interface's slot/port location on all or most of the network devices in the service.

**Step 4** Choose the CE Encapsulation type.

The choices are:

- **FRAME RELAY**
- **FRAME RELAY IETF**

**Note** If the Interface Type is ANY, ISC will not ask for an **Encapsulation** type in the policy.

**Step 5** Check the **UNI Shutdown** check box if you want to leave the UNI port shut during service activation, for example, when the service provider wants to deploy a service in the network but wants to activate it at a later time.

**Step 6** Check the **PW Tunnel Selection** check box if you want to be able to manually select the Traffic Engineering (TE) tunnel for the pseudowire connecting point-to-point N-PEs.

This attribute is unchecked by default

Subsequently, when you create a service request based on this policy, you must specify the TE tunnel ID in a field provided. ISC uses the tunnel information to create and provision a pseudowire class that describes the pseudowire connection between two N-PEs. This pseudowire class can be shared by more than one pseudowire, as long as the pseudowires share the same tunnel ID and remote loopback address. You are responsible to ensure that the tunnel interface and associated ID are configured. During service request creation when you specify the tunnel ID number, ISC does not check the validity of the value. That is, ISC does not verify the existence of the tunnel.

**Step 7** Click the **Next** button, if you want to enable template support for the policy.

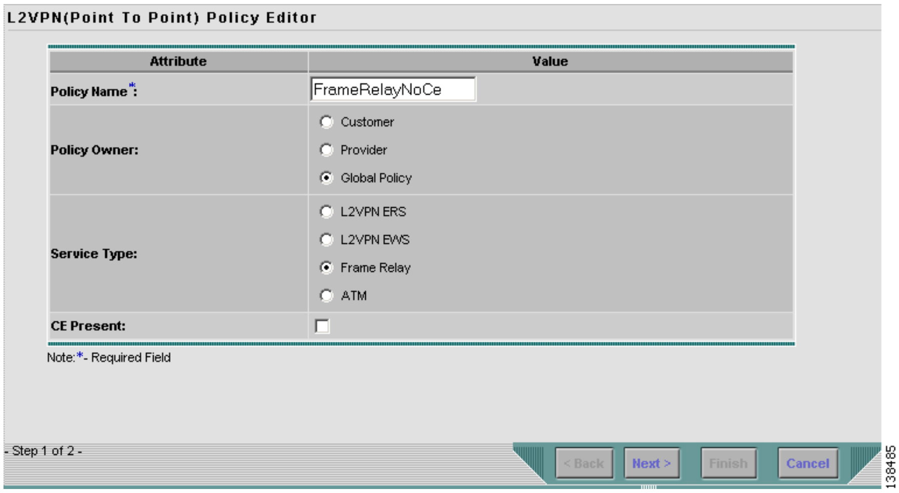
The Template Association window appears. In this window, you can enable template support and, optionally, associate templates and data files with the policy. For instructions about associating templates with policies and how to use the features in this window, see Appendix B, “Working with Templates and Data Files”. When you have completed setting up templates and data files for the policy, click **Finish** in the Template Association window to close it and return to the Policy Editor window.

**Step 8** Click **Finish**.

# Defining a Frame Relay Policy without a CE

This section describes how to define a Frame Relay policy without a CE present. Figure 7-22 is an example of the first page of this policy.

***Figure 7-22 Frame Relay Policy without a CE***



L2VPN(Point To Point) Policy Editor

| Attribute | Value |
|---|---|
| Policy Name*: | FrameRelayNoCe |
| Policy Owner: | ○ Customer<br>○ Provider<br>● Global Policy |
| Service Type: | ○ L2VPN ERS<br>○ L2VPN EWS<br>● Frame Relay<br>○ ATM |
| CE Present: | □ |

Note:*- Required Field

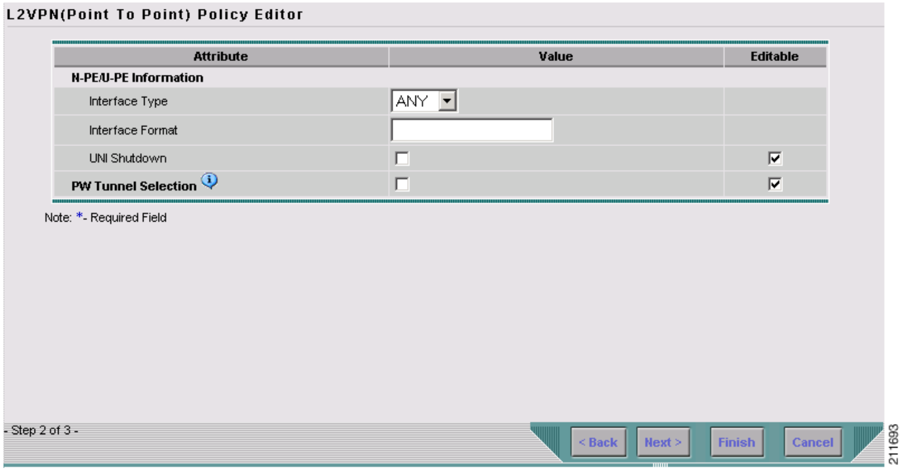
- Step 1 of 2 -

Perform the following steps.

**Step 1** Click **Next**. The window in Figure 7-23 appears.

The **Editable** check box gives you the option of making a field editable. If you check the **Editable** check box, the service operator who is using this L2VPN policy can modify the editable parameter during L2VPN service request creation.

***Figure 7-23 Frame Relay without CE Policy Attributes***



L2VPN(Point To Point) Policy Editor

| Attribute | Value | Editable |
|---|---|---|
| N-PE/U-PE Information | | |
| Interface Type | ANY | |
| Interface Format | | |
| UNI Shutdown | □ | ☑ |
| PW Tunnel Selection | □ | ☑ |

Note: *- Required Field

- Step 2 of 3 -

**Step 2** Choose the N-PE/U-PE **Interface Type** for the **CE** from the drop-down list.

The choices are:

- **ANY**
- **Serial**
- **MFR**
- **POS**
- **Hssi**
- **BRI**

**Step 3** Enter an **Interface Format** as the slot number/port number for the PE interface (for example, **1/0** indicates that the interface is located at slot 1, port 0).

This is especially useful to specify here if you know that the link will always go through a particular interface's slot/port location on all or most of the network devices in the service.

**Step 4** Choose the N-PE/U-PE **Encapsulation** type.

The choices are:

- **FRAME RELAY**
- **FRAME RELAY IETF**

**Note** If the Interface Type is ANY, ISC will not ask for an **Encapsulation** type in the policy.

**Step 5** Check the **UNI Shutdown** check box if you want to leave the UNI port shut during service activation, for example, when the service provider wants to deploy a service in the network but wants to activate it at a later time.

**Step 6** Check the **PW Tunnel Selection** check box if you want to be able to manually select the Traffic Engineering (TE) tunnel for the pseudowire connecting point-to-point N-PEs.

This attribute is unchecked by default

Subsequently, when you create a service request based on this policy, you must specify the TE tunnel ID in a field provided. ISC uses the tunnel information to create and provision a pseudowire class that describes the pseudowire connection between two N-PEs. This pseudowire class can be shared by more than one pseudowire, as long as the pseudowires share the same tunnel ID and remote loopback address. You are responsible to ensure that the tunnel interface and associated ID are configured. During service request creation when you specify the tunnel ID number, ISC does not check the validity of the value. That is, ISC does not verify the existence of the tunnel.

**Step 7** Click the **Next** button, if you want to enable template support for the policy.

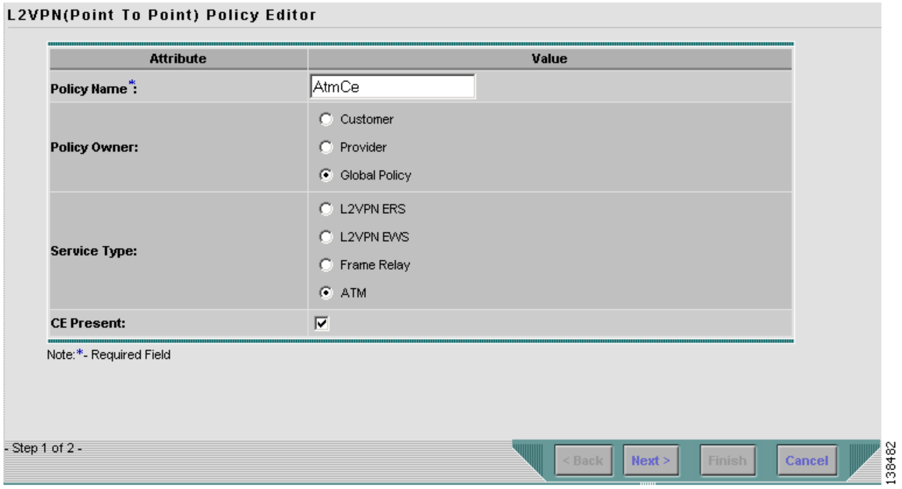
The Template Association window appears. In this window, you can enable template support and, optionally, associate templates and data files with the policy. For instructions about associating templates with policies and how to use the features in this window, see Appendix B, "Working with Templates and Data Files". When you have completed setting up templates and data files for the policy, click **Finish** in the Template Association window to close it and return to the Policy Editor window.

**Step 8** Click **Finish**.

# Defining an ATM Policy with a CE

This section describes how to define an ATM policy with CE present. Figure 7-24 is an example of the first page of this policy.

*Figure 7-24 ATM Policy with a CE*



L2VPN(Point To Point) Policy Editor

| Attribute | Value |
|---|---|
| Policy Name *: | AtmCe |
| Policy Owner: | ○ Customer<br>○ Provider<br>● Global Policy |
| Service Type: | ○ L2VPN ERS<br>○ L2VPN EWS<br>○ Frame Relay<br>● ATM |
| CE Present: | ☑ |

Note: *- Required Field

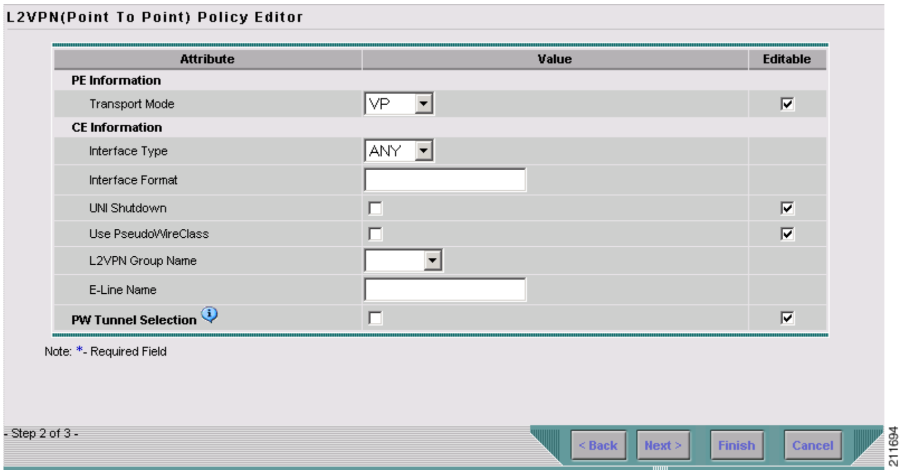
- Step 1 of 2 -

Perform the following steps.

**Step 1** Click **Next**. The window in Figure 7-25 appears.

The **Editable** check box gives you the option of making a field editable. If you check the **Editable** check box, the service operator who is using this L2VPN policy can modify the editable parameter during L2VPN service request creation.

*Figure 7-25 ATM with CE Policy Attributes*



L2VPN(Point To Point) Policy Editor

| Attribute | Value | Editable |
|---|---|---|
| PE Information | | |
| Transport Mode | VP | ☑ |
| CE Information | | |
| Interface Type | ANY | |
| Interface Format | | |
| UNI Shutdown | ☐ | ☑ |
| Use PseudoWireClass | ☐ | ☑ |
| L2VPN Group Name | | |
| E-Line Name | | |
| PW Tunnel Selection | ☐ | ☑ |

Note: *- Required Field

- Step 2 of 3 -

**Step 2** Choose the **Transport Mode** from the drop-down list.

The choices are:

- **VP**—Virtual path mode. This is the default.
- **VC**—Virtual circuit mode.
- **PORT**—Port mode. (Only supported for the IOS XR 3.7 platform.)

**Note** If you choose **PORT** as the transport mode, the attributes **ATM VCD/Sub-interface #** and **ATM VPI** will be disabled in the Link Attributes window of the service request based on this policy.

**Step 3** Choose the **CE Interface Type** from the drop-down list.

The choices are:

- **ANY**
- **ATM**
- **Switch**

**Step 4** Enter an **Interface Format** as the slot number/port number for the CE interface (for example, **1/0** indicates that the interface is located at slot 1, port 0).

This is especially useful to specify here if you know that the link will always go through a particular interface's slot/port location on all or most of the network devices in the service.

**Step 5** Choose a **CE Encapsulation**.

The choices are:

- **AAL5SNAP**
- **AAL5MUX**
- **AAL5NLPID**
- **AAL2**

**Note** If the Interface Type is ANY, ISC will not ask for an **Encapsulation** type in the policy.

**Step 6** Check the **UNI Shutdown** check box if you want to leave the UNI port shut during service activation, for example, when the service provider wants to deploy a service in the network but wants to activate it at a later time.

**Step 7** Check the **Use PseudoWireClass** check box to enable the selection of a pseudowire class.

This attribute is only applicable for IOS XR devices. If the check box is checked, an additional attribute, **PseudoWireClass**, appears in the GUI. Click the **Select** button of PseudoWireClass attribute to choose a pseudowire class previously created in ISC. The pseudowire class name is used for provisioning pw-class commands on IOS XR devices. See Creating and Modifying Pseudowire Classes for IOS XR Devices, page 2-10 for additional information on pseudowire class support for IOS XR devices.

**Step 8** Choose an **L2VPN Group Name** from the drop-down list.

The choices are:

- **ISC**
- **VPNSC**

This attribute is used for providing the L2VPN group name on IOS XR devices.

**Note** The choices in the drop-down list are derived from a configurable DCPL property. For information about how to define the L2VPN Group Name choices available in the drop-down list, see Defining L2VPN Group Names for IOS XR Devices, page 2-14.

**Step 9** Enter an **E-Line Name** to specify the point-to-point (p2p) E-line name.

This attribute is only applicable for IOS XR devices. If no value is specified for the p2p name, ISC generates a default name consisting of the names of the two PEs forming the pseudowire, separated by hyphens (for example, 6503-A----6503-B). If the default name is more than 32 characters, the device names are truncated.

**Step 10** Check the **PW Tunnel Selection** check box if you want to be able to manually select the Traffic Engineering (TE) tunnel for the pseudowire connecting point-to-point N-PEs.

This attribute is unchecked by default

Subsequently, when you create a service request based on this policy, you must specify the TE tunnel ID in a field provided. ISC uses the tunnel information to create and provision a pseudowire class that describes the pseudowire connection between two N-PEs. This pseudowire class can be shared by more than one pseudowire, as long as the pseudowires share the same tunnel ID and remote loopback address. You are responsible to ensure that the tunnel interface and associated ID are configured. During service request creation when you specify the tunnel ID number, ISC does not check the validity of the value. That is, ISC does not verify the existence of the tunnel.

**Step 11** Click the **Next** button, if you want to enable template support for the policy.

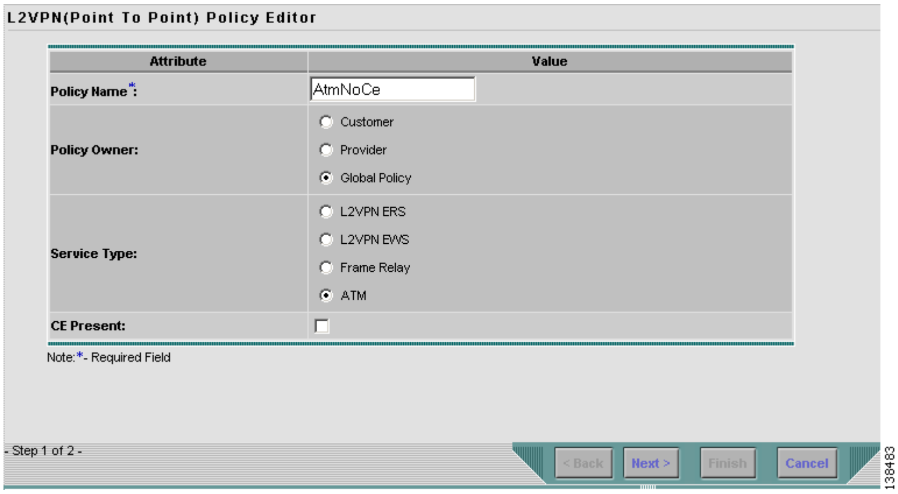
The Template Association window appears. In this window, you can enable template support and, optionally, associate templates and data files with the policy. For instructions about associating templates with policies and how to use the features in this window, see Appendix B, "Working with Templates and Data Files". When you have completed setting up templates and data files for the policy, click **Finish** in the Template Association window to close it and return to the Policy Editor window.

**Step 12** Click **Finish**.

# Defining an ATM Policy without a CE

This section describes how to define an ATM policy without a CE present. Figure 7-26 is an example of the first page of this policy.

*Figure 7-26 ATM Policy without a CE*



L2VPN(Point To Point) Policy Editor

| Attribute | Value |
|---|---|
| Policy Name*: | AtmNoCe |
| Policy Owner: | ○ Customer<br>○ Provider<br>● Global Policy |
| Service Type: | ○ L2VPN ERS<br>○ L2VPN EWS<br>○ Frame Relay<br>● ATM |
| CE Present: | □ |

Note:*- Required Field

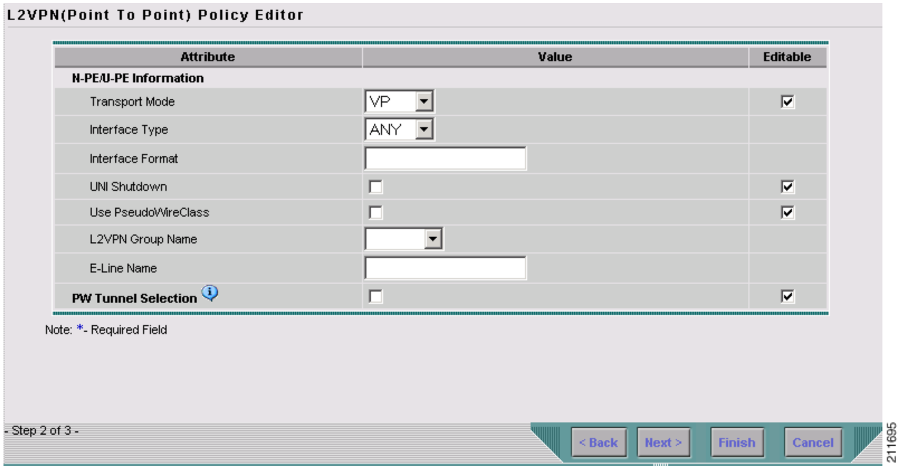
- Step 1 of 2 -

Perform the following steps.

**Step 1** Click **Next**. The window in Figure 7-27 appears.

The **Editable** check box gives you the option of making a field editable. If you check the **Editable** check box, the service operator who is using this L2VPN policy can modify the editable parameter during L2VPN service request creation.

*Figure 7-27 ATM without CE Policy Attributes*



L2VPN(Point To Point) Policy Editor

| Attribute | Value | Editable |
|---|---|---|
| N-PE/U-PE Information | | |
| Transport Mode | VP | ☑ |
| Interface Type | ANY | |
| Interface Format | | |
| UNI Shutdown | □ | ☑ |
| Use PseudoWireClass | □ | ☑ |
| L2VPN Group Name | | |
| E-Line Name | | |
| PW Tunnel Selection | □ | ☑ |

Note: *- Required Field

- Step 2 of 3 -

**Step 2** Choose the **Transport Mode** from the drop-down list.

The choices are:

- **VP**—Virtual path mode. This is the default.
- **VC**—Virtual circuit mode.
- **PORT**—Port mode. (Only supported for the IOS XR 3.7 platform.)

**Note** If you choose **PORT** as the transport mode, the attributes **ATM VCD/Sub-interface #** and **ATM VPI** will be disabled in the Link Attributes window of the service request based on this policy.

**Step 3** Choose the **N-PE/U-PE Interface Type** from the drop-down list.

The choices are:

- **ANY**
- **ATM**
- **Switch**

**Step 4** Enter an **Interface Format** as the slot number/port number for the PE interface (for example, **1/0** indicates that the interface is located at slot 1, port 0).

This is especially useful to specify here if you know that the link will always go through a particular interface's slot/port location on all or most of the network devices in the service.

**Step 5** Choose a **PE Encapsulation**.

The choices are:

- **AAL5**
- **AAL0**

**Note** If the Interface Type is ANY, ISC will not ask for an **Encapsulation** type in the policy.

**Step 6** Check the **UNI Shutdown** check box if you want to leave the UNI port shut during service activation, for example, when the service provider wants to deploy a service in the network but wants to activate it at a later time.

**Step 7** Check the **Use PseudoWireClass** check box to enable the selection of a pseudowire class.

This attribute is only applicable for IOS XR devices. If the check box is checked, an additional attribute, **PseudoWireClass**, appears in the GUI. Click the **Select** button of PseudoWireClass attribute to choose a pseudowire class previously created in ISC. The pseudowire class name is used for provisioning pw-class commands on IOS XR devices. See Creating and Modifying Pseudowire Classes for IOS XR Devices, page 2-10 for additional information on pseudowire class support for IOS XR devices.

**Step 8** Choose an **L2VPN Group Name** from the drop-down list.

The choices are:

- **ISC**
- **VPNSC**

This attribute is used for provisioning the L2VPN group name on IOS XR devices.

**Note** The choices in the drop-down list are derived from a configurable DCPL property. For information about how to define the L2VPN Group Name choices available in the drop-down list, see Defining L2VPN Group Names for IOS XR Devices, page 2-14.

**Step 9** Enter an **E-Line Name** to specify the point-to-point (p2p) E-line name.

This attribute is only applicable for IOS XR devices. If no value is specified for the p2p name, ISC generates a default name consisting of the names of the two PEs forming the pseudowire, separated by hyphens (for example, 6503-A----6503-B). If the default name is more than 32 characters, the device names are truncated.

**Step 10** Check the **PW Tunnel Selection** check box if you want to be able to manually select the Traffic Engineering (TE) tunnel for the pseudowire connecting point-to-point N-PEs.

This attribute is unchecked by default

Subsequently, when you create a service request based on this policy, you must specify the TE tunnel ID in a field provided. ISC uses the tunnel information to create and provision a pseudowire class that describes the pseudowire connection between two N-PEs. This pseudowire class can be shared by more than one pseudowire, as long as the pseudowires share the same tunnel ID and remote loopback address. You are responsible to ensure that the tunnel interface and associated ID are configured. During service request creation when you specify the tunnel ID number, ISC does not check the validity of the value. That is, ISC does not verify the existence of the tunnel.

**Step 11** Click the **Next** button, if you want to enable template support for the policy.

The Template Association window appears. In this window, you can enable template support and, optionally, associate templates and data files with the policy. For instructions about associating templates with policies and how to use the features in this window, see Appendix B, "Working with Templates and Data Files". When you have completed setting up templates and data files for the policy, click **Finish** in the Template Association window to close it and return to the Policy Editor window.

**Step 12** Click **Finish**.

CHAPTER 8

# Managing an L2VPN Service Request

This chapter covers the basic steps to provision an ERS (EVPL), EWS (EPL), ATM, or Frame Relay L2VPN service. It contains the following sections:

## Introducing L2VPN Service Requests

An L2VPN service request consists of one or more end-to-end wires, connecting various sites in a point-to-point topology. When you create a service request, you enter several parameters, including the specific interfaces on the CE and PE routers. You can also associate Cisco IP Solution Center (ISC) templates and data files with a service request. See Appendix B, "Working with Templates and Data Files," for more about using templates and data files in service requests.

To create a service request, a Service Policy must already be defined, as described in Chapter 7, "Creating an L2VPN Policy."

Based on the predefined L2VPN policy, an operator creates an L2VPN service request, with or without modifications to the L2VPN policy, and deploys the service. Service creation and deployment are normally performed by regular network technicians for daily operation of network provisioning.

**Note** Not all of the attributes defined in an L2VPN policy might be applicable to a service request. For specific information, see L2VPN policy attribute descriptions in Chapter 7, "Creating an L2VPN Policy".

The following steps are involved in creating a service request for Layer 2 connectivity between customer sites:

- Choose a CE Topology for ERS (EVPL)/Frame Relay/ATM services.
- Choose the endpoints (CE and PE) that must be connected. For each end-to-end Layer 2 connection, ISC creates an end-to-end wire object in the repository for the service request.
- Choose a CE or PE interface.
- Choose a Named Physical Circuit (NPC) for the CE or PE.
- Edit the end-to-end connection.
- Edit the link attributes.
- (0ptional) Associate templates and data files to devices in the service request.

# Creating an L2VPN Service Request

To create an L2VPN service request, perform the following steps.

**Step 1** Choose **Service Inventory > Inventory and Connection Manage > Service Requests**.

The Service Requests window appears.

**Step 2** Click **Create**.

**Step 3** Choose **L2VPN** from the drop-down list.

L2VPN service requests must be associated with an L2VPN policy. You choose an L2VPN policy from the policies previously created (see Chapter 7, "Creating an L2VPN Policy").

**Step 4** Choose the L2VPN policy of choice.

If more than one L2VPN policy exists, a list of L2VPN policies appears.

**Step 5** When you make the choice, click **OK**.

As soon as you make the choice, the new service request inherits all the properties of that L2VPN policy, such as all the editable and non-editable features and pre-set parameters.

To continue creating an L2VPN service request, go to one of the following sections:

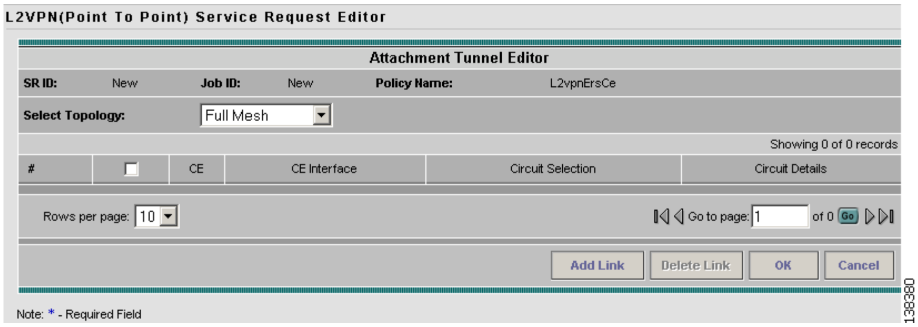
- Creating an ERS (EVPL), ATM, or Frame Relay L2VPN Service Request with a CE, page 8-2.
- Creating an EWS (EPL) L2VPN Service Request with a CE, page 8-8.
- Creating an ERS (EVPL), ATM, or Frame Relay L2VPN Service Request without a CE, page 8-11.
- Creating an EWS (EPL) L2VPN Service Request without a CE, page 8-16.

# Creating an ERS (EVPL), ATM, or Frame Relay L2VPN Service Request with a CE

This section includes detailed steps for creating an L2VPN service request with a CE present for ERS (EVPL), ATM, and Frame Relay policies. If you are creating an L2VPN service request for an EWS (EPL) policy, go to Creating an EWS (EPL) L2VPN Service Request with a CE, page 8-8.

After you choose an L2VPN policy, the L2VPN Service Request Editor window appears. (See Figure 8-1.)

*Figure 8-1 L2VPN Service Request Editor*



L2VPN(Point To Point) Service Request Editor

Attachment Tunnel Editor

SR ID: New Job ID: New Policy Name: L2vpnErsCe

Select Topology: Full Mesh

Showing 0 of 0 records

| # | □ | CE | CE Interface | Circuit Selection | Circuit Details |
|---|---|---|---|---|---|

Rows per page: 10 Go to page: 1 of 0

Note: * - Required Field

Perform the following steps.

**Step 1** Choose a **Topology** from the drop-down list. If you choose **Full Mesh**, each CE will have direct connections to every other CE.

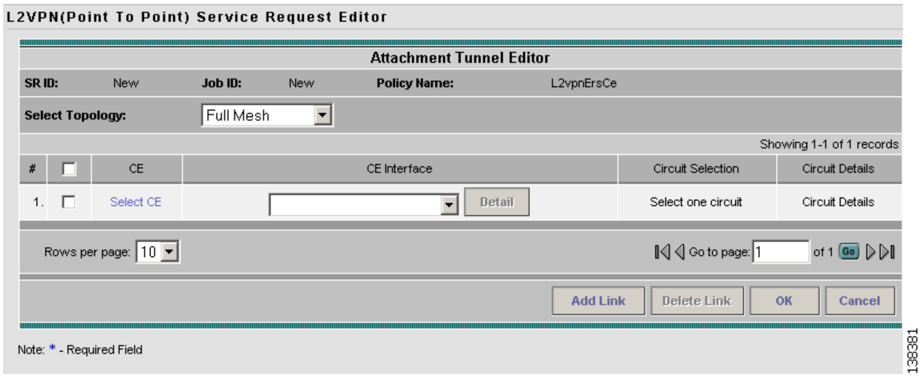
If you choose **Hub and Spoke**, then only the Hub CE has connection to each Spoke CE and the Spoke CEs do not have direct connection to each other.

**Note** The full mesh and the hub and spoke topologies make a difference only when you choose more than two endpoints. For example, with four endpoints, ISC automatically creates six links with full mesh topology. With hub and spoke topology, however, ISC creates only three links.

**Step 2** Click **Add Link**.

You specify the CE endpoints using the Attachment Tunnel Editor. You can create one or more CEs from a window like the one in Figure 8-2.

*Figure 8-2 Select CE*



L2VPN(Point To Point) Service Request Editor

Attachment Tunnel Editor

SR ID: New Job ID: New Policy Name: L2vpnErsCe

Select Topology: Full Mesh

Showing 1-1 of 1 records

| # | □ | CE | CE Interface | Circuit Selection | Circuit Details |
|---|---|---|---|---|---|
| 1. | □ | Select CE | Detail | Select one circuit | Circuit Details |

Rows per page: 10 Go to page: 1 of 1

Note: * - Required Field

**Note** All the services that deploy point-to-point connections (ERS/EVPL, EWS/EPL, ATMoMPLS, and FRoMPLS) must have at least two CEs specified.

**Step 3** Click **Select CE** in the CE column.

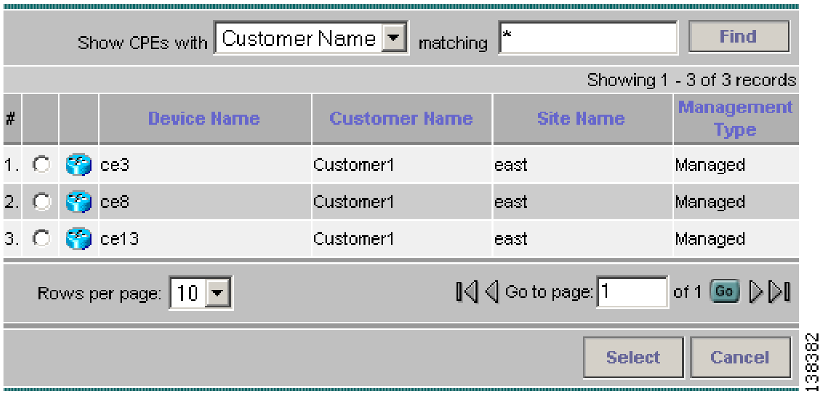
The CPE for Attachment Circuit window appears. (See Figure 8-3.) This window displays the list of currently defined CEs.

a. From the **Show CPEs with** drop-down list, you can display CEs by Customer Name, by Site, or by Device Name.

b. You can use the **Find** button to either search for a specific CE, or to refresh the display.

c. You can set the **Rows per page** to 5, 10, 20, 30, 40, or All.

***Figure 8-3 Select CPE Device***



Show CPEs with Customer Name matching *

Showing 1 - 3 of 3 records

| # | | | Device Name | Customer Name | Site Name | Management Type |
|---|---|---|---|---|---|---|
| 1. | ○ | | ce3 | Customer1 | east | Managed |
| 2. | ○ | | ce8 | Customer1 | east | Managed |
| 3. | ○ | | ce13 | Customer1 | east | Managed |

Rows per page: 10

Go to page: 1 of 1

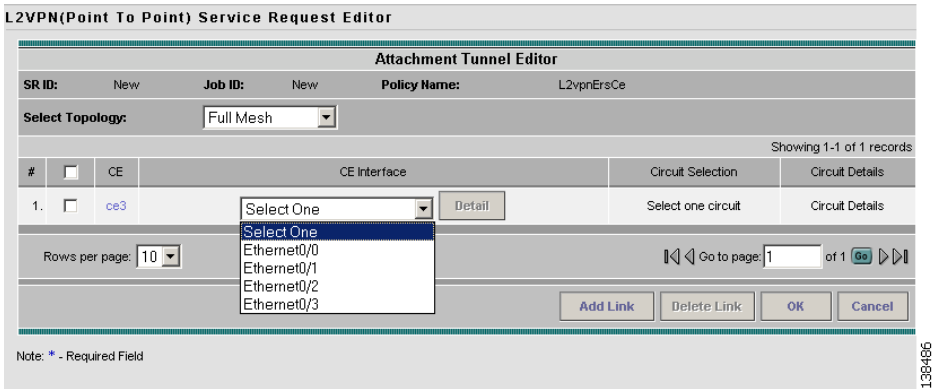
**Step 4** In the Select column, choose a CE for the L2VPN link.

**Step 5** Click **Select**.

The Service Request Editor window appears displaying the name of the selected CE in the CE column.

**Step 6** Choose the CE interface from the drop-down list. (See Figure 8-4.)

***Figure 8-4 Select the CE Interface***



L2VPN(Point To Point) Service Request Editor

Attachment Tunnel Editor

SR ID: New Job ID: New Policy Name: L2vpnErsCe

Select Topology: Full Mesh

Showing 1-1 of 1 records

| # | □ | CE | CE Interface | Circuit Selection | Circuit Details |
|---|---|---|---|---|---|
| 1. | □ | ce3 | Select One | Select one circuit | Circuit Details |

Rows per page: 10

Go to page: 1 of 1

Note: * - Required Field

**Note** When you provision an L2VPN ERS (EVPL) service, when you choose a UNI for a particular device, ISC determines if there are other services using the same UNI. If so, a warning message is displayed. If you ignore the message and save the service request, all of the underlying service requests relying on the same UNI are synchronized with the modified shared attributes of the latest service request. In addition, the state of the existing service requests is changed to the Requested state.

**Note** ISC only displays the available interfaces for the service, based on the configuration of the underlying interfaces, existing service requests that might be using the interface, and the customer associated with the service request. You can click the **Details** button to display a pop-up window with information on the available interfaces, such as interface name, customer name, VPN name and service request ID, service request type, VLAN translation type, and VLAN ID information.

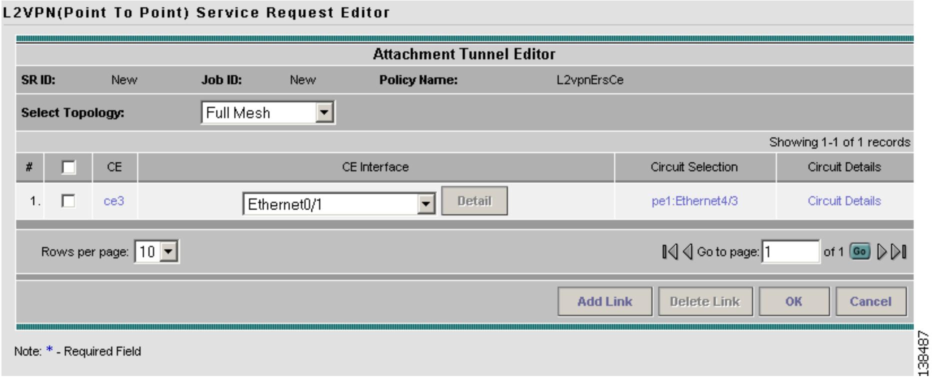
**Step 7** If only one NPC exists for the Chosen CE and CE interface, that NPC is autopopulated in the Circuit Selection column and you need not choose it explicitly. If more then one NPC is available, click **Select one circuit** in the Circuit Selection column.

The NPC window appears, enabling you to choose the appropriate NPC.

**Step 8** Click **OK**.

Each time you choose a CE and its interface, the NPC that was precreated from this CE and interface is automatically displayed under **Circuit Selection**. (See Figure 8-5.) This means that you do not have to further specify the PE to complete the link.

*Figure 8-5 NPC Created*



L2VPN(Point To Point) Service Request Editor

Attachment Tunnel Editor

SR ID: New Job ID: New Policy Name: L2vpnErsCe

Select Topology: Full Mesh

Showing 1-1 of 1 records

| # | ☐ | CE | CE Interface | Circuit Selection | Circuit Details |
|---|---|---|---|---|---|
| 1. | ☐ | ce3 | Ethernet0/1 Detail | pe1:Ethernet4/3 | Circuit Details |

Rows per page: 10

Go to page: 1 of 1

Note: * - Required Field

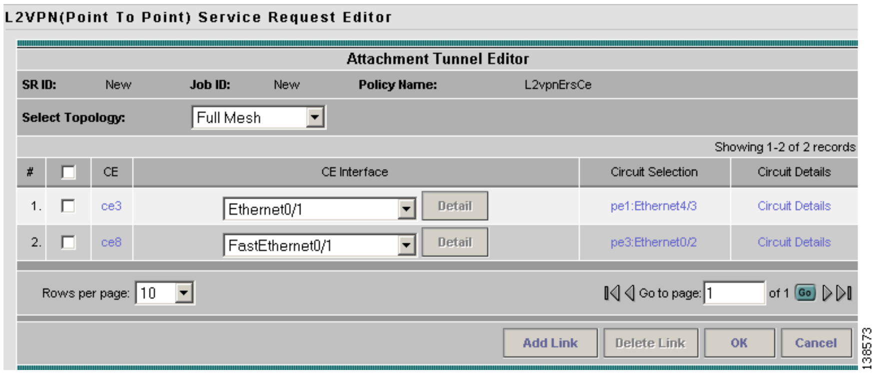
If you want to review the details of this NPC, click **Circuit Details** in the Circuit Details column. The NPC Details window appears and lists the circuit details for this NPC.

**Step 9** Continue to specify additional CEs, as in previous steps.

ISC creates the links between CEs based on the Topology that you chose.

**Step 10** Click **OK** in Figure 8-6.

*Figure 8-6 NPCs Created*



L2VPN(Point To Point) Service Request Editor

Attachment Tunnel Editor

SR ID: New Job ID: New Policy Name: L2vpnErsCe

Select Topology: Full Mesh

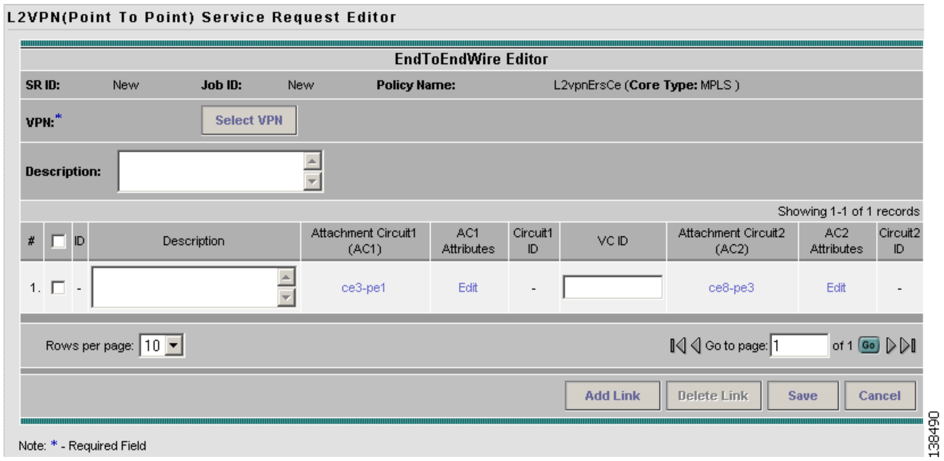
Showing 1-2 of 2 records

| # | □ | CE | CE Interface | | Circuit Selection | Circuit Details |
|---|---|---|---|---|---|---|
| 1. | □ | ce3 | Ethernet0/1 | Detail | pe1:Ethernet4/3 | Circuit Details |
| 2. | □ | ce8 | FastEthernet0/1 | Detail | pe3:Ethernet0/2 | Circuit Details |

Rows per page: 10 Go to page: 1 of 1 Go

For ERS (EVPL), ATM, and Frame Relay, the End-to-End Wire Editor window appears. (See Figure 8-7.)

*Figure 8-7 End-to-End Wire Editor*



L2VPN(Point To Point) Service Request Editor

EndToEndWire Editor

SR ID: New Job ID: New Policy Name: L2vpnErsCe (Core Type: MPLS )

VPN:* Select VPN

Description:

Showing 1-1 of 1 records

| # | □ | ID | Description | Attachment Circuit1 (AC1) | AC1 Attributes | Circuit1 ID | VC ID | Attachment Circuit2 (AC2) | AC2 Attributes | Circuit2 ID |
|---|---|---|---|---|---|---|---|---|---|---|
| 1. | □ | - | | ce3-pe1 | Edit | - | | ce8-pe3 | Edit | - |

Rows per page: 10 Go to page: 1 of 1 Go

Note: * - Required Field

**Step 11** The VPN for this service request appears in the **VPN** field.

If there is more than one VPN, click **Select VPN** to choose a VPN. The VPN for L2VPN service request window appears.

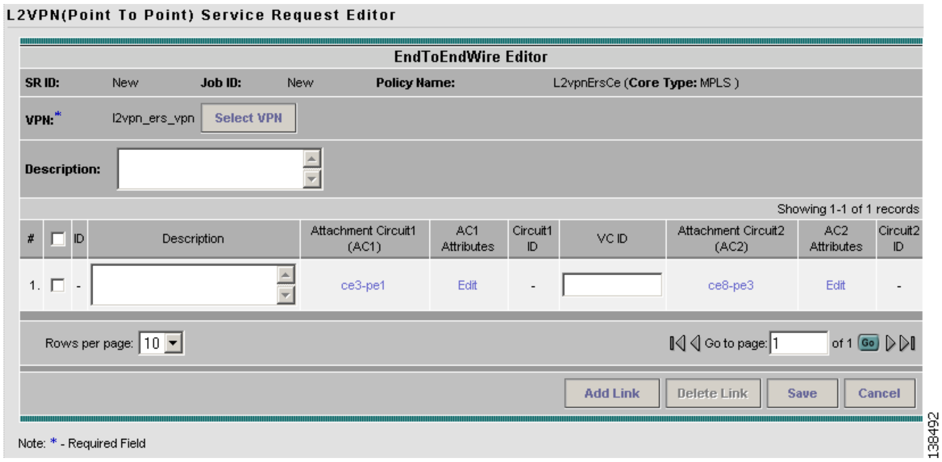
**Step 12** Choose a **VPN Name** and click **Select**.

The L2VPN Service Request Editor window appears with the VPN name displayed.

**Step 13** If necessary, click **Add AC** in the Attachment Circuit AC2 column, and repeat Steps 3 to 10 for AC2.

The End-to-End Wire Editor window displays the complete end-to-end wire. (See Figure 8-8.)

*Figure 8-8 End-to-End Wire Created*



L2VPN(Point To Point) Service Request Editor

EndToEndWire Editor

| | | | | | |
|---|---|---|---|---|---|
| **SR ID:** | New | **Job ID:** | New | **Policy Name:** | L2vpnErsCe (**Core Type:** MPLS ) |
| **VPN:*** | l2vpn_ers_vpn | Select VPN | | | |
| **Description:** | | | | | |

Showing 1-1 of 1 records

| # | □ | ID | Description | Attachment Circuit1 (AC1) | AC1 Attributes | Circuit1 ID | VC ID | Attachment Circuit2 (AC2) | AC2 Attributes | Circuit2 ID |
|---|---|---|---|---|---|---|---|---|---|---|
| 1. | □ | - | | ce3-pe1 | Edit | - | | ce8-pe3 | Edit | - |

Rows per page: 10

Go to page: 1 of 1 Go

Add Link | Delete Link | Save | Cancel

Note: * - Required Field

**Step 14** Specify remaining items in the End-to-End-Wire Editor window as necessary for your configuration:

- You can choose any of the blue highlighted values to edit the End-to-End Wire.
- You can edit the AC link attributes to change the default policy settings. After you edit these fields, the blue link changes from Default to Changed. For more information, see the section Modifying the L2VPN Service Request, page 8-20.
- You can enter a description for the service request in the first **Description** field. The description will show up in this window and also in the Description column of the Service Requests window. The maximum length for this field is 256 characters.
- You can enter a description for each end-to-end wire in the **Description** field provided for each wire. The description shows up only in this window. The data in this field is not pushed to the device(s). The maximum length for this field is 256 characters.
- The ID number is system-generated identification number for the circuit.
- The Circuit ID is created automatically, based on the service. For example, for Ethernet, it is based on the VLAN number; for Frame Relay, it is based on the DLCI; for ATM, it is based on the VPI/VCI.
- If the policy was set up for you to define a VC ID manually, enter it into the empty **VC ID** field. If policy was set to "auto pick" the VC ID, ISC will supply a VC ID, and this field will not be editable. In the case where you supply the VC ID manually, if the entered value is in the provider's range, ISC validates if the entered value is available or allocated. If the entered value has been already allocated, ISC generates an error message saying that the entered value is not available and prompts you to re-enter the value. If the entered value is in the provider's range, and if it is available, then it is allocated and is removed from the VC ID pool. If the entered value is outside the provider's range, ISC displays a warning saying that no validation could be performed to verify if it is available or allocated.
- You can also click **Add Link** to add an end-to-end wire.
- You can click **Delete Link** to delete an end-to-end wire.

**Step 15** When you are finished editing the end-to-end wires, click **Save**.

The service request is created and saved into ISC.

# Creating an EWS (EPL) L2VPN Service Request with a CE

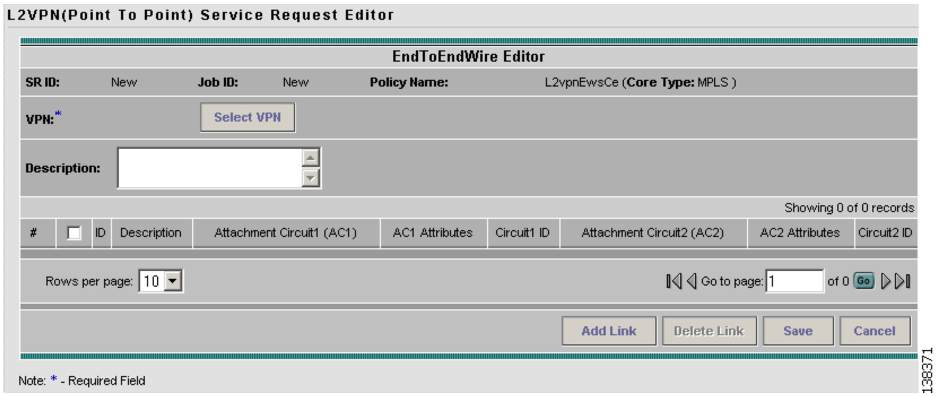
This section includes detailed steps for creating an L2VPN service request with a CE present for EWS (EPL). If you are creating an L2VPN service request for an ERS (EVPL), ATM, or Frame Relay policy, go to Creating an ERS (EVPL), ATM, or Frame Relay L2VPN Service Request with a CE, page 8-2.

Perform the following steps.

**Step 1** Create the L2VPN service request for EWS (EPL) with CE.

The L2VPN Service Request Editor window appears. (See Figure 8-9.)

***Figure 8-9 EWS (EPL) Service Request Editor***



L2VPN(Point To Point) Service Request Editor

EndToEndWire Editor

**SR ID:** New **Job ID:** New **Policy Name:** L2vpnEwsCe (**Core Type:** MPLS )

**VPN:*** Select VPN

**Description:**

Showing 0 of 0 records

| # | □ | ID | Description | Attachment Circuit1 (AC1) | AC1 Attributes | Circuit1 ID | Attachment Circuit2 (AC2) | AC2 Attributes | Circuit2 ID |
|---|---|---|---|---|---|---|---|---|---|

Rows per page: 10 Go to page: 1 of 0

Note: * - Required Field

**Step 2** Click **Select VPN** to choose a VPN for use with this CE.

The Select VPN window appears with the VPNs defined in the system.

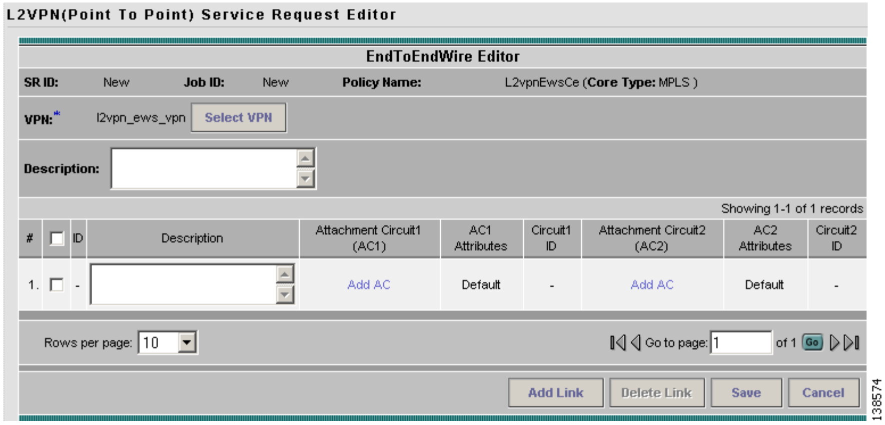
**Step 3** Choose a **VPN Name** in the Select column.

**Step 4** Click **Select**.

The L2VPN Service Request Editor window appears with the VPN name displayed.

**Step 5** Click **Add Link**. (See Figure 8-10.)

*Figure 8-10 End-To-End Wire Editor*



L2VPN(Point To Point) Service Request Editor

EndToEndWire Editor

**SR ID:** New **Job ID:** New **Policy Name:** L2vpnEwsCe (**Core Type:** MPLS )

**VPN:*** l2vpn_ews_vpn Select VPN

**Description:**

Showing 1-1 of 1 records

| # | □ ID | Description | Attachment Circuit1 (AC1) | AC1 Attributes | Circuit1 ID | Attachment Circuit2 (AC2) | AC2 Attributes | Circuit2 ID |
|---|---|---|---|---|---|---|---|---|
| 1. | □ - | | Add AC | Default | - | Add AC | Default | - |

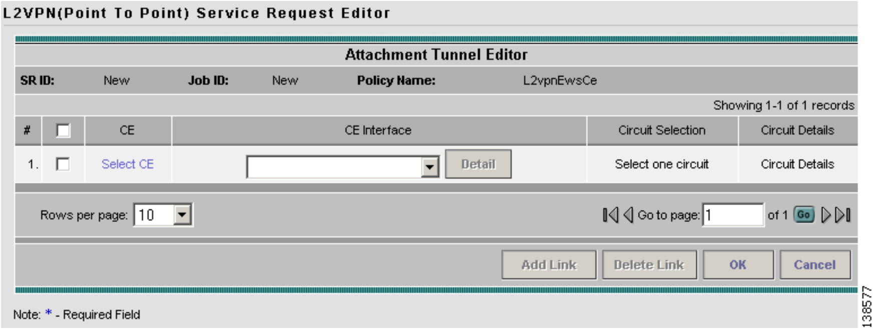
Rows per page: 10 Go to page: 1 of 1 Go

- You can enter a description for the service request in the first **Description** field. The description will show up in this window and also in the Description column of the Service Requests window. The maximum length for this field is 256 characters.
- You can enter a description for each end-to-end wire in the **Description** field provided for each wire. The description shows up only in this window. The data in this field is not pushed to the device(s). The maximum length for this field is 256 characters.
- The ID number is system-generated identification number for the circuit.
- The Circuit ID is created automatically, based on the service. For example, for Ethernet, it is based on the VLAN number; for Frame Relay, it is based on the DLCI; for ATM, it is based on the VPI/VCI.

**Step 6** Click **Add AC** in the Attachment Circuit (A1) column.

The Attachment Tunnel Editor appears. (See Figure 8-11.)

*Figure 8-11 Select CE for Attachment Circuit*



L2VPN(Point To Point) Service Request Editor

Attachment Tunnel Editor

**SR ID:** New **Job ID:** New **Policy Name:** L2vpnEwsCe

Showing 1-1 of 1 records

| # | □ | CE | CE Interface | Circuit Selection | Circuit Details |
|---|---|---|---|---|---|
| 1. | □ | Select CE | Detail | Select one circuit | Circuit Details |

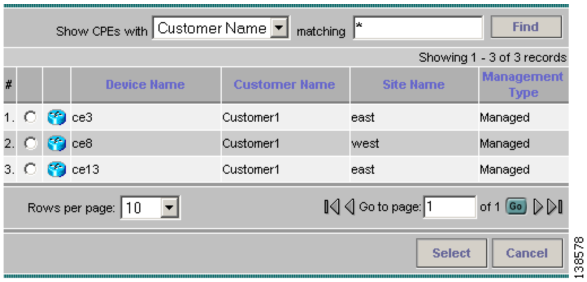
Rows per page: 10 Go to page: 1 of 1 Go

Note: * - Required Field

**Step 7** Click **Select CE**.

The Select CPE window appears. (See Figure 8-12.)

*Figure 8-12 CPE for Attachment Circuit*



Show CPEs with Customer Name ▼ matching *

Showing 1 - 3 of 3 records

| # | | Device Name | Customer Name | Site Name | Management Type |
|---|---|---|---|---|---|
| 1. | ○ | ce3 | Customer1 | east | Managed |
| 2. | ○ | ce8 | Customer1 | west | Managed |
| 3. | ○ | ce13 | Customer1 | east | Managed |

Rows per page: 10 ▼ Go to page: 1 of 1

This window displays the list of currently defined CEs.

a. From the **Show CPEs with** drop-down list, you can display CEs by Customer Name, by Site, or by Device Name.

b. You can use the **Find** button to either search for a specific CE, or to refresh the display.

c. You can set the **Rows per page** to 5, 10, 20, 30, 40, or All.

**Step 8** In the Select column, choose a CE for the L2VPN link.

**Step 9** Click **Select**.

**Step 10** In the Attachment Tunnel Editor window, choose a CE interface from the drop-down list.

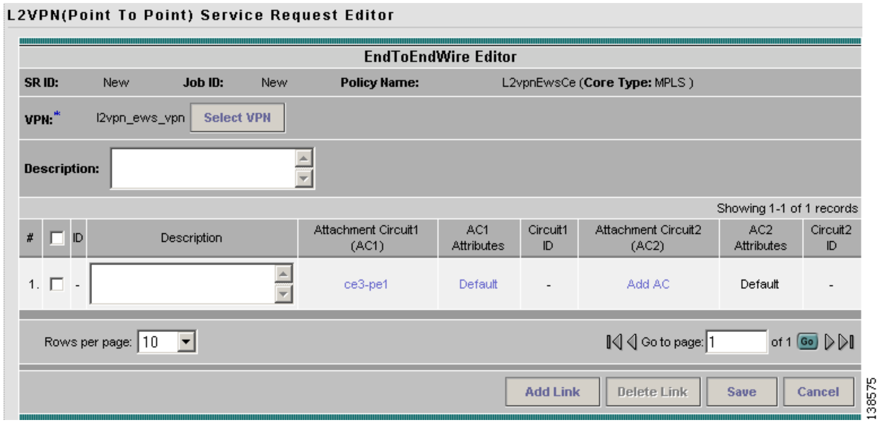
**Step 11** If only one NPC exists for the Chosen CE and CE interface, that NPC is autopopulated in the Circuit Selection column and you need not choose it explicitly.

If more then one NPC is available, click **Select one circuit** in the Circuit Selection column. The NPC window appears, enabling you to choose the appropriate NPC. Each time you choose a CE and its interface, the NPC that was precreated from this CE and interface is automatically displayed under **Circuit Selection**. This means that you do not have to further specify the PE to complete the link.

**Step 12** Click **OK**.

The EndToEndWire Editor window appears displaying the name of the selected CE in the AC1 column. (See Figure 8-13.)

*Figure 8-13 NPC Created*



L2VPN(Point To Point) Service Request Editor

EndToEndWire Editor

SR ID: New Job ID: New Policy Name: L2vpnEwsCe (Core Type: MPLS )

VPN:* l2vpn_ews_vpn

Description:

Showing 1-1 of 1 records

| # | ☐ | ID | Description | Attachment Circuit1 (AC1) | AC1 Attributes | Circuit1 ID | Attachment Circuit2 (AC2) | AC2 Attributes | Circuit2 ID |
|---|---|---|---|---|---|---|---|---|---|
| 1. | ☐ | - | | ce3-pe1 | Default | - | Add AC | Default | - |

Rows per page: 10 ▼ Go to page: 1 of 1

**Step 13** Click **AC1** Link Attributes and edit the attributes if desired.

For more information, see the section Modifying the L2VPN Service Request, page 8-20.

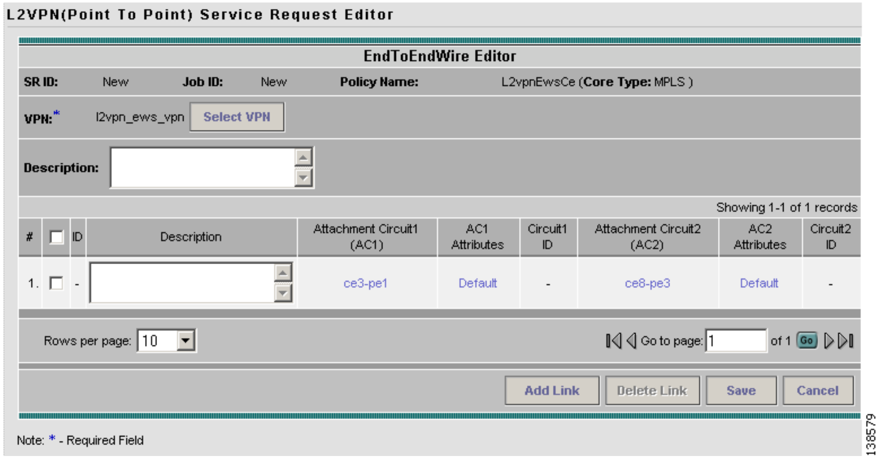
**Step 14** Click **OK**.

**Step 15** Repeat Steps 6 through 14 for **AC2**.

**Step 16** Click **OK**.

You see a window like Figure 8-14.

***Figure 8-14 Attachment Circuits Selected***



L2VPN(Point To Point) Service Request Editor

EndToEndWire Editor

**SR ID:** New **Job ID:** New **Policy Name:** L2vpnEwsCe (**Core Type:** MPLS )

**VPN:*** l2vpn_ews_vpn

**Description:**

Showing 1-1 of 1 records

| # | ☐ | ID | Description | Attachment Circuit1 (AC1) | AC1 Attributes | Circuit1 ID | Attachment Circuit2 (AC2) | AC2 Attributes | Circuit2 ID |
|---|---|---|---|---|---|---|---|---|---|
| 1. | ☐ | - | | ce3-pe1 | Default | - | ce8-pe3 | Default | - |

Rows per page: 10

Go to page: 1 of 1

Note: * - Required Field

**Step 17** Click **Save**.

The EWS (EPL) service request is created and saved in ISC.

# Creating an ERS (EVPL), ATM, or Frame Relay L2VPN Service Request without a CE

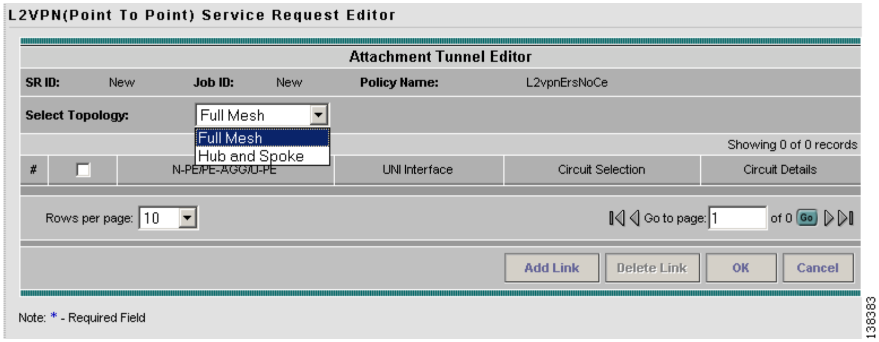
This section includes detailed steps for creating an L2VPN service request without a CE present for ERS (EVPL), ATM, and Frame Relay policies. If you are creating an L2VPN service request for an EWS (EPL) policy, go to Creating an EWS (EPL) L2VPN Service Request without a CE, page 8-16.

Perform the following steps.

**Step 1** Create the L2VPN service request for ERS (EVPL) without a CE.

The L2VPN Service Request Editor window appears. (See Figure 8-15.)

*Figure 8-15 L2VPN Service Request Editor*



L2VPN(Point To Point) Service Request Editor

Attachment Tunnel Editor

SR ID: New Job ID: New Policy Name: L2vpnErsNoCe

Select Topology: Full Mesh

Showing 0 of 0 records

| # | ☐ | N-PE/PE-AGG/U-PE | UNI Interface | Circuit Selection | Circuit Details |
|---|---|---|---|---|---|

Rows per page: 10

Go to page: 1 of 0

Note: * - Required Field

**Step 2** Choose a **Topology** from the drop-down list.

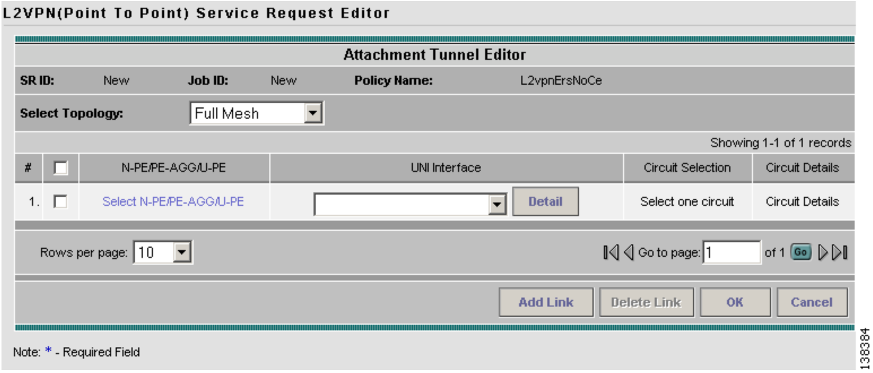
If you choose **Full Mesh**, each CE will have direct connections to every other CE. If you choose **Hub and Spoke**, then only the Hub CE has connection to each Spoke CE and the Spoke CEs do not have direct connection to each other.

**Note** The full mesh and the hub and spoke topologies make a difference only when you choose more than two endpoints. For example, with four endpoints, ISC automatically creates six links with full mesh topology. With hub and spoke topology, however, ISC creates only three links.

**Step 3** Click **Add Link**.

The Attachment Tunnel Editor window appears. (See Figure 8-16.)

*Figure 8-16 Select U-PE/PE-AGG/N-PE*



L2VPN(Point To Point) Service Request Editor

Attachment Tunnel Editor

SR ID: New Job ID: New Policy Name: L2vpnErsNoCe

Select Topology: Full Mesh

Showing 1-1 of 1 records

| # | ☐ | N-PE/PE-AGG/U-PE | UNI Interface | Circuit Selection | Circuit Details |
|---|---|---|---|---|---|
| 1. | ☐ | Select N-PE/PE-AGG/U-PE | Detail | Select one circuit | Circuit Details |

Rows per page: 10

Go to page: 1 of 1

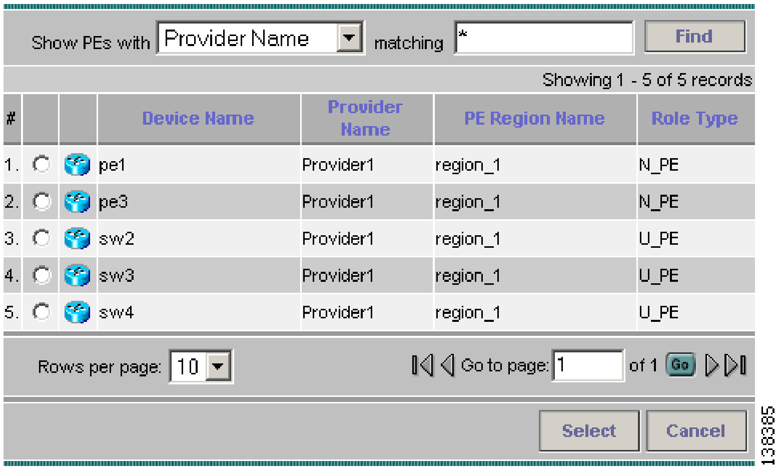
Note: * - Required Field

**Step 4** Specify the N-PE/PE-AGG/U-PE endpoints using the Attachment Tunnel Editor, as covered in the following steps.

**Step 5** Click **Select U-PE/PE-AGG/N-PE** in the U-PE/PE-AGG/N-PE column.

The PE for Attachment Circuit window appears. (See Figure 8-17).

*Figure 8-17 Select PE Device*



Show PEs with Provider Name ▾ matching * Find

Showing 1 - 5 of 5 records

| # | | | Device Name | Provider Name | PE Region Name | Role Type |
|---|---|---|---|---|---|---|
| 1. | ○ | | pe1 | Provider1 | region_1 | N_PE |
| 2. | ○ | | pe3 | Provider1 | region_1 | N_PE |
| 3. | ○ | | sw2 | Provider1 | region_1 | U_PE |
| 4. | ○ | | sw3 | Provider1 | region_1 | U_PE |
| 5. | ○ | | sw4 | Provider1 | region_1 | U_PE |

Rows per page: 10 ▾ Go to page: 1 of 1 Go

This window displays the list of currently defined PEs.

**a.** The **Show PEs with** drop-down list shows PEs by customer name, by site, or by device name.

**b.** The **Find** button allows a search for a specific PE or a refresh of the window.

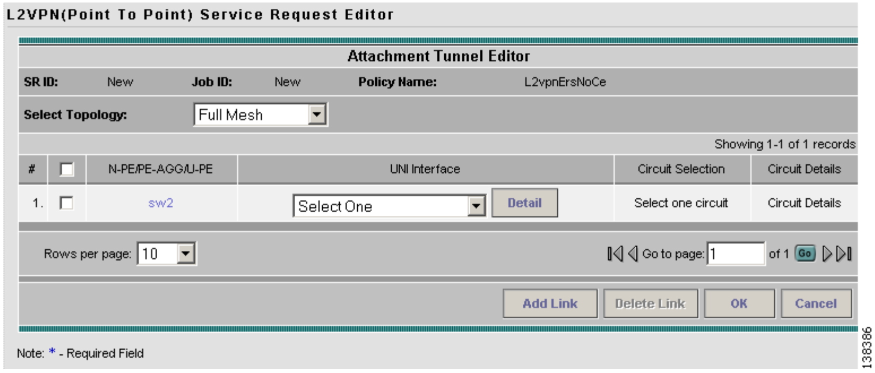
**c.** The **Rows per page** drop-down list allows the page to be set to 5, 10, 20, 30, 40, or All.

**Step 6** In the **Select** column, choose the PE device name for the L2VPN link.

**Step 7** Click **Select**.

The Service Request Editor window appears displaying the name of the selected PE in the PE column. (See Figure 8-18.)

*Figure 8-18 Select the UNI Interface*



L2VPN(Point To Point) Service Request Editor

Attachment Tunnel Editor

SR ID: New Job ID: New Policy Name: L2vpnErsNoCe

Select Topology: Full Mesh ▾

Showing 1-1 of 1 records

| # | □ | N-PE/PE-AGG/U-PE | UNI Interface | Circuit Selection | Circuit Details |
|---|---|---|---|---|---|
| 1. | □ | sw2 | Select One ▾ Detail | Select one circuit | Circuit Details |

Rows per page: 10 ▾ Go to page: 1 of 1 Go

Note: * - Required Field

**Step 8** Choose the UNI interface from the drop-down list.

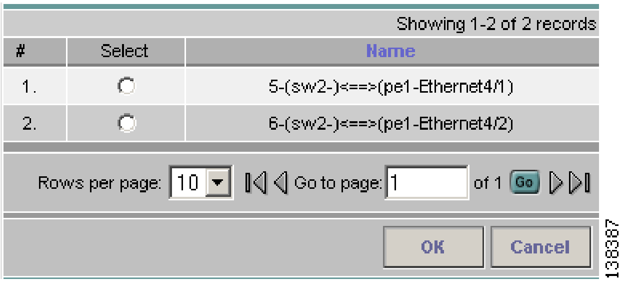
**Note** When you provision an L2VPN ERS (EVPL) service, when you choose a UNI for a particular device, ISC determines if there are other services using the same UNI. If so, a warning message is displayed. If you ignore the message and save the service request, all of the underlying service requests lying on the same UNI are synchronized with the modified shared attributes of the latest service request. In addition, the state of the existing service requests is changed to the Requested state.

**Note** ISC only displays the available interfaces for the service, based on the configuration of the underlying interfaces, existing service requests that might be using the interface, and the customer associated with the service request. You can click the **Details** button to display a pop-up window with information on the available interfaces, such as interface name, customer name, VPN name and service request ID, service request type, VLAN translation type, and VLAN ID information.

**Step 9** If the PE role type is U-PE, click **Select one circuit** in the Circuit Selection column.

The NPC window appears. (See Figure 8-19.)

***Figure 8-19 Select NPC***



Showing 1-2 of 2 records

| # | Select | Name |
|---|---|---|
| 1. | ○ | 5-(sw2-)<==>(pe1-Ethernet4/1) |
| 2. | ○ | 6-(sw2-)<==>(pe1-Ethernet4/2) |

Rows per page: 10 Go to page: 1 of 1

If only one NPC exists for the Chosen PE and PE interface, that NPC is auto populated in the Circuit Selection column and you need not choose it explicitly.

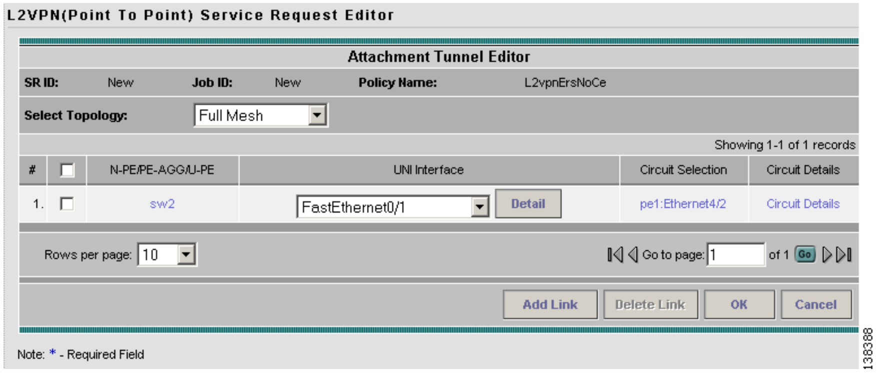
**Note** If the PE role type is N-PE, the columns Circuit Selection and Circuit Details are disabled.

**Step 10** Choose the name of the NPC from the **Select** column.

**Step 11** Click **OK**.

Each time you choose a PE and its interface, the NPC that was precreated from this PE and interface is automatically displayed under **Circuit Selection**. (See Figure 8-20.) This means that you do not have to further specify the PE to complete the link.

***Figure 8-20 NPC Created***



L2VPN(Point To Point) Service Request Editor

Attachment Tunnel Editor

SR ID: New Job ID: New Policy Name: L2vpnErsNoCe

Select Topology: Full Mesh

Showing 1-1 of 1 records

| # | □ | N-PE/PE-AGG/U-PE | UNI Interface | Circuit Selection | Circuit Details |
|---|---|---|---|---|---|
| 1. | □ | sw2 | FastEthernet0/1 Detail | pe1:Ethernet4/2 | Circuit Details |

Rows per page: 10 Go to page: 1 of 1

Note: * - Required Field

**Step 12** If you want to review the details of this NPC, click **Circuit Details** in the Circuit Details column.

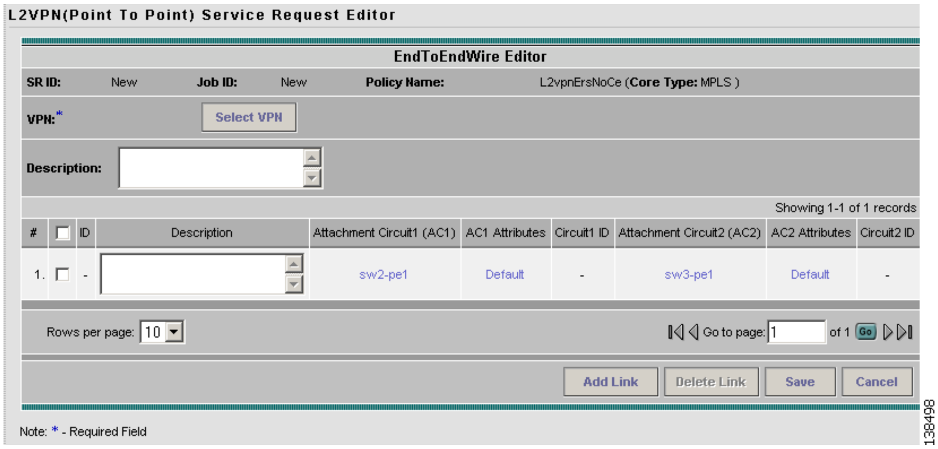
The NPC Details window appears and lists the circuit details for this NPC.

**Step 13** After you specify all the PEs, ISC creates the links between PEs based on the Topology that you chose.

**Step 14** Click **OK**.

For ERS (EVPL), ATM, and Frame Relay, the End-to-End-Wire Editor window appears. (See Figure 8-21.)

***Figure 8-21 End-to-End Wire Editor***



L2VPN(Point To Point) Service Request Editor

EndToEndWire Editor

SR ID: New Job ID: New Policy Name: L2vpnErsNoCe (**Core Type:** MPLS )

VPN:* Select VPN

Description:

Showing 1-1 of 1 records

| # | □ | ID | Description | Attachment Circuit1 (AC1) | AC1 Attributes | Circuit1 ID | Attachment Circuit2 (AC2) | AC2 Attributes | Circuit2 ID |
|---|---|---|---|---|---|---|---|---|---|
| 1. | □ | - | | sw2-pe1 | Default | - | sw3-pe1 | Default | - |

Rows per page: 10 Go to page: 1 of 1 Go

Note: * - Required Field

**Step 15** The VPN for this service request appears in the Select VPN field.

If there is more than one VPN, click **Select VPN** to choose a VPN.

**Step 16** Specify remaining items in the End-to-End-Wire Editor window, as necessary for your configuration:

- You can choose any of the blue highlighted values to edit the End-to-End Wire.
- You can edit the AC link attributes to change the default policy settings. After you edit these fields, the blue link changes from Default to Changed. For more information, see the section Modifying the L2VPN Service Request, page 8-20.
- You can also click **Add Link** to add an end-to-end wire.
- You can click **Delete Link** to delete an end-to-end wire.

**Note** If you are attempting to decommission a service request to which a template has been added, see Monitoring Service Requests, page 11-10 for information on the proper way to do this.

- You can enter a description for the service request in the first **Description** field. The description will show up in this window and also in the Description column of the Service Requests window. The maximum length for this field is 256 characters.
- You can enter a description for each end-to-end wire in the **Description** field provided for each wire. The description shows up only in this window. The data in this field is not pushed to the device(s). The maximum length for this field is 256 characters.
- The ID number is system-generated identification number for the circuit.
- The Circuit ID is created automatically, based on the service. For example, for Ethernet, it is based on the VLAN number; for Frame Relay, it is based on the DLCI; for ATM, it is based on the VPI/VCI.

**Step 17** When you are finished editing the end-to-end wires, click **Save**.

The service request is created and saved into ISC.

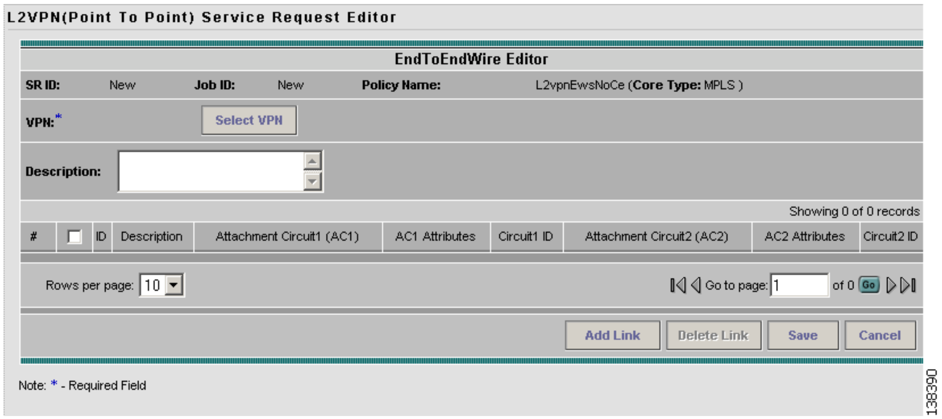
# Creating an EWS (EPL) L2VPN Service Request without a CE

This section includes detailed steps for creating an L2VPN service request without a CE present for EWS (EPL). If you are creating an L2VPN service request for an ERS (EVPL), ATM, or Frame Relay policy, see Creating an ERS (EVPL), ATM, or Frame Relay L2VPN Service Request without a CE, page 8-11.

**Step 1** Create the L2VPN service request for EWS (EPL) without a CE.

The L2VPN Service Request Editor window appears. (See Figure 8-22.)

*Figure 8-22 EWS (EPL) Service Request Editor*



L2VPN(Point To Point) Service Request Editor

EndToEndWire Editor

**SR ID:** New **Job ID:** New **Policy Name:** L2vpnEwsNoCe (**Core Type:** MPLS )

**VPN:***

**Description:**

Showing 0 of 0 records

| # | □ | ID | Description | Attachment Circuit1 (AC1) | AC1 Attributes | Circuit1 ID | Attachment Circuit2 (AC2) | AC2 Attributes | Circuit2 ID |
|---|---|---|---|---|---|---|---|---|---|

Rows per page: 10

Go to page: 1 of 0

Note: * - Required Field

**Step 2** Click **Select VPN** to choose a VPN for use with this PE.

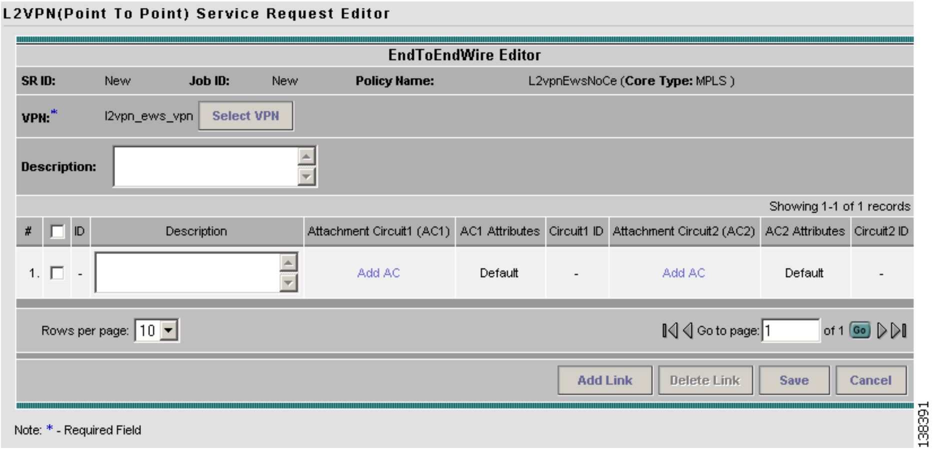
The Select a VPN window appears with the VPNs defined in the system.

**Step 3** Choose a **VPN Name** in the Select column.

**Step 4** Click **Select**.

The End-To-End Wire Editor window appears with the VPN name displayed. (See Figure 8-23.)

*Figure 8-23 End-To-End Wire Editor*



L2VPN(Point To Point) Service Request Editor

EndToEndWire Editor

**SR ID:** New **Job ID:** New **Policy Name:** L2vpnEwsNoCe (**Core Type:** MPLS )

**VPN:*** l2vpn_ews_vpn [Select VPN]

**Description:**

Showing 1-1 of 1 records

| # | □ | ID | Description | Attachment Circuit1 (AC1) | AC1 Attributes | Circuit1 ID | Attachment Circuit2 (AC2) | AC2 Attributes | Circuit2 ID |
|---|---|---|---|---|---|---|---|---|---|
| 1. | □ | - |  | Add AC | Default | - | Add AC | Default | - |

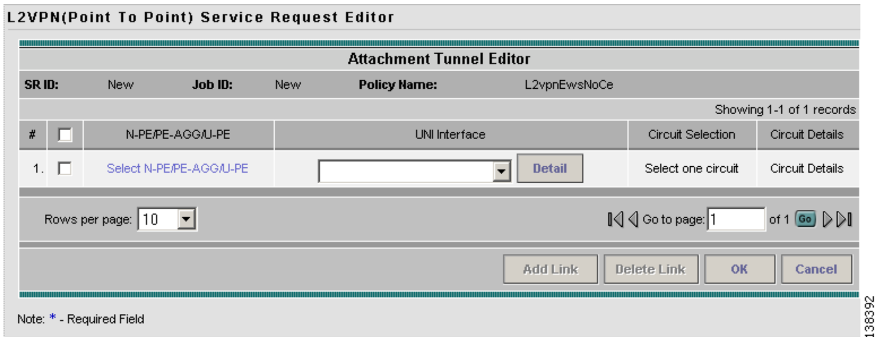
Rows per page: 10 Go to page: 1 of 1 Go

Note: * - Required Field

**Step 5** Click **Add AC** in the Attachment Circuit (AC1) column.

The Attachment Tunnel Editor window appears. (See Figure 8-24.)

*Figure 8-24 Select the PE for the Attachment Circuit*



L2VPN(Point To Point) Service Request Editor

Attachment Tunnel Editor

**SR ID:** New **Job ID:** New **Policy Name:** L2vpnEwsNoCe

Showing 1-1 of 1 records

| # | □ | N-PE/PE-AGG/U-PE | UNI Interface | Circuit Selection | Circuit Details |
|---|---|---|---|---|---|
| 1. | □ | Select N-PE/PE-AGG/U-PE | [Detail] | Select one circuit | Circuit Details |

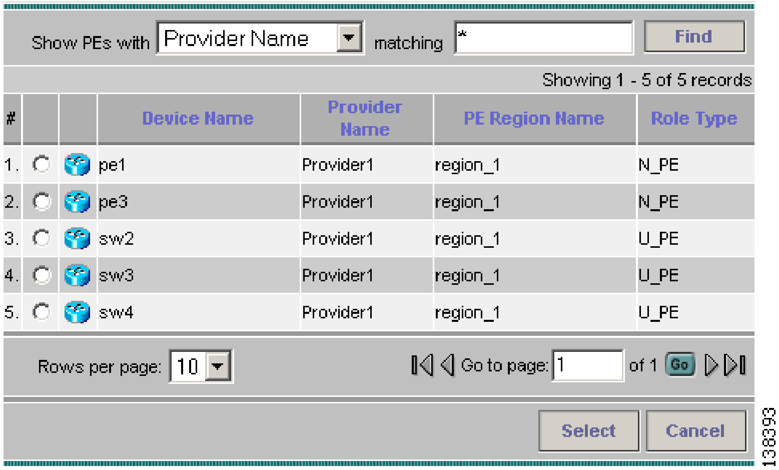
Rows per page: 10 Go to page: 1 of 1 Go

Note: * - Required Field

**Step 6** Click **Select N-PE/PE-AGG/U-PE**.

The Select PE Device window appears. (See Figure 8-25.)

*Figure 8-25 PE for Attachment Circuit*



Show PEs with Provider Name matching *  Find

Showing 1 - 5 of 5 records

| # | | | Device Name | Provider Name | PE Region Name | Role Type |
|---|---|---|---|---|---|---|
| 1. | ○ | | pe1 | Provider1 | region_1 | N_PE |
| 2. | ○ | | pe3 | Provider1 | region_1 | N_PE |
| 3. | ○ | | sw2 | Provider1 | region_1 | U_PE |
| 4. | ○ | | sw3 | Provider1 | region_1 | U_PE |
| 5. | ○ | | sw4 | Provider1 | region_1 | U_PE |

Rows per page: 10  Go to page: 1 of 1 Go

This window displays the list of currently defined PEs.

a. From the **Show PEs with** drop-down list, you can display PEs by Customer Name, by Site, or by Device Name.

b. You can use the **Find** button to either search for a specific PE, or to refresh the display.

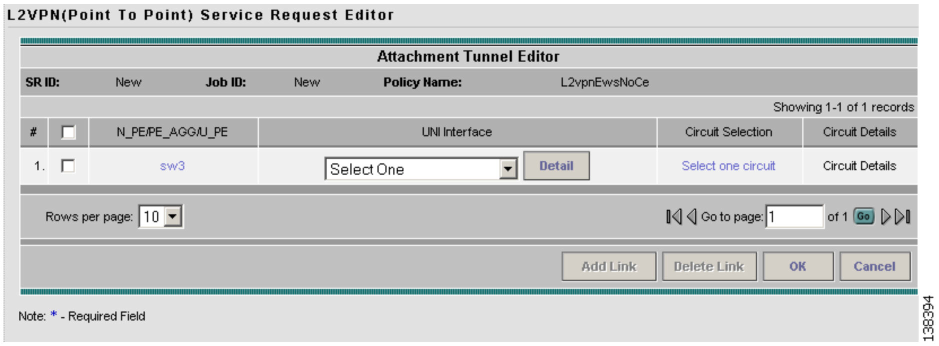
c. You can set the **Rows per page** to 5, 10, 20, 30, 40, or All.

**Step 7** In the Select column, choose a PE for the L2VPN link.

**Step 8** Click **Select**.

The Attachment Tunnel Editor window appears. (See Figure 8-26.)

*Figure 8-26 PE Interface*



L2VPN(Point To Point) Service Request Editor

Attachment Tunnel Editor

**SR ID:** New **Job ID:** New **Policy Name:** L2vpnEwsNoCe

Showing 1-1 of 1 records

| # | □ | N_PE/PE_AGG/U_PE | UNI Interface | Circuit Selection | Circuit Details |
|---|---|---|---|---|---|
| 1. | □ | sw3 | Select One  Detail | Select one circuit | Circuit Details |

Rows per page: 10  Go to page: 1 of 1 Go

Note: * - Required Field

**Step 9** Choose a PE interface from the drop-down list.

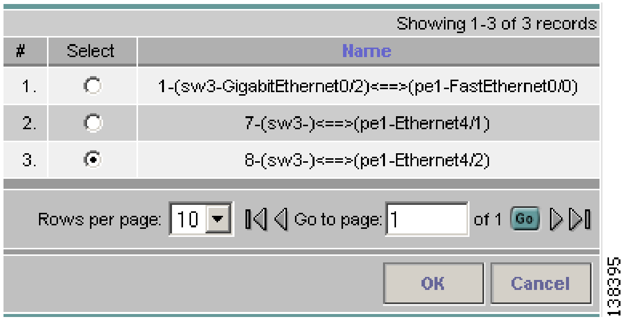
**Note** ISC only displays the available interfaces for the service, based on the configuration of the underlying interfaces, existing service requests that might be using the interface, and the customer associated with the service request. You can click the **Details** button to display a pop-up window with information on the available interfaces, such as interface name, customer name, VPN name and service request ID, service request type, VLAN translation type, and VLAN ID information.

**Step 10** If the PE role type is N-PE, the columns Circuit Selection and Circuit Details are disabled. In this case, skip to Step 13.

**Step 11** If the PE role type is U-PE, click **Select one circuit** in the Circuit Selection column.

The Select NPC window appears. (See Figure 8-27.)

*Figure 8-27 Select NPC*



Showing 1-3 of 3 records

| # | Select | Name |
|---|---|---|
| 1. | ○ | 1-(sw3-GigabitEthernet0/2)<==>(pe1-FastEthernet0/0) |
| 2. | ○ | 7-(sw3-)<==>(pe1-Ethernet4/1) |
| 3. | ● | 8-(sw3-)<==>(pe1-Ethernet4/2) |

Rows per page: 10 Go to page: 1 of 1

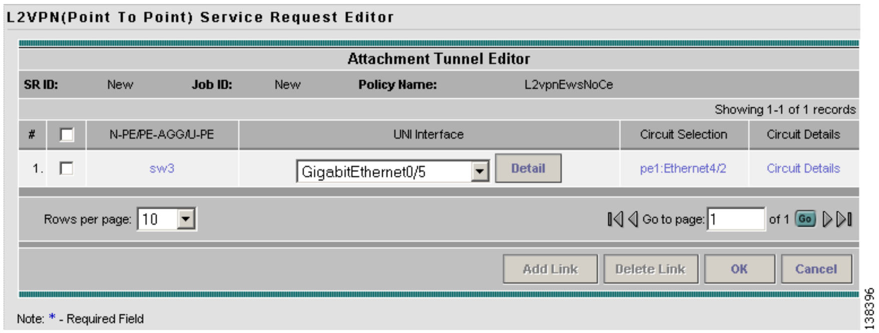
**Note** If only one NPC exists for the Chosen PE and PE interface, that NPC is auto populated in the Circuit Selection column and you need not choose it explicitly.

**Step 12** If applicable, choose the name of the NPC from the Select column.

**Step 13** Click **OK**.

The Attachment Tunnel Editor appears. (See Figure 8-28.)

*Figure 8-28 NPC Created*



L2VPN(Point To Point) Service Request Editor

Attachment Tunnel Editor

**SR ID:** New **Job ID:** New **Policy Name:** L2vpnEwsNoCe

Showing 1-1 of 1 records

| # | □ | N-PE/PE-AGG/U-PE | UNI Interface | Circuit Selection | Circuit Details |
|---|---|---|---|---|---|
| 1. | □ | sw3 | GigabitEthernet0/5 Detail | pe1:Ethernet4/2 | Circuit Details |

Rows per page: 10 Go to page: 1 of 1

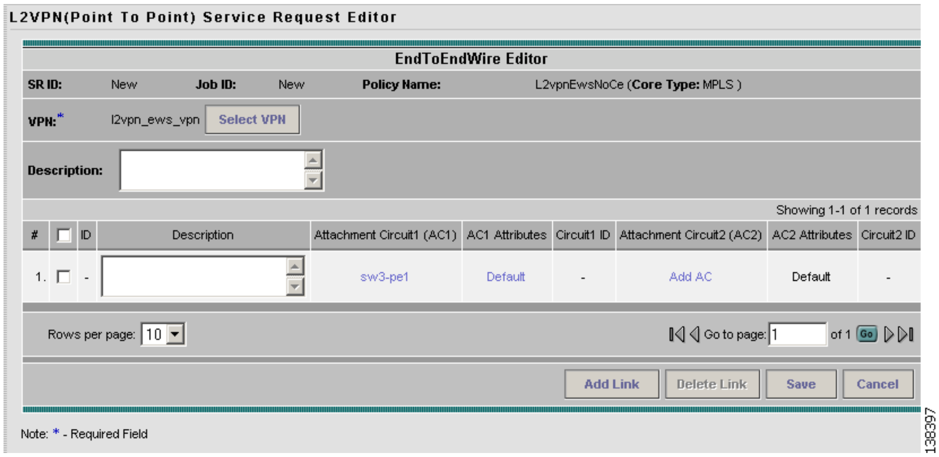
Note: * - Required Field

**Note** Each time you choose a PE and its interface, the NPC that was precreated from this PE and interface is automatically displayed under **Circuit Selection**. (See Figure 8-28.) This means that you do not have to further specify the PE to complete the link.

**Step 14** Click **OK**.

The Service Request Editor window appears displaying the name of the selected PE in the AC1 column. (See Figure 8-29.)

*Figure 8-29 Attachment Circuit Selected*



L2VPN(Point To Point) Service Request Editor

EndToEndWire Editor

SR ID: New Job ID: New Policy Name: L2vpnEwsNoCe (Core Type: MPLS )

VPN:* l2vpn_ews_vpn

Description:

Showing 1-1 of 1 records

| # | □ | ID | Description | Attachment Circuit1 (AC1) | AC1 Attributes | Circuit1 ID | Attachment Circuit2 (AC2) | AC2 Attributes | Circuit2 ID |
|---|---|---|---|---|---|---|---|---|---|
| 1. | □ | - | | sw3-pe1 | Default | - | Add AC | Default | - |

Rows per page: 10

Go to page: 1 of 1

Note: * - Required Field

**Step 15** Click **AC1** Link Attributes and edit the attributes, if desired.

For more information, see the section Modifying the L2VPN Service Request, page 8-20.

**Step 16** Repeat Steps 5 through 14 for **AC2**.

**Step 17** Specify remaining items in the End-to-End-Wire Editor window, as necessary for your configuration.

- You can enter a description for the service request in the first **Description** field. The description will show up in this window and also in the Description column of the Service Requests window. The maximum length for this field is 256 characters.
- You can enter a description for each end-to-end wire in the **Description** field provided for each wire. The description shows up only in this window. The data in this field is not pushed to the device(s). The maximum length for this field is 256 characters.
- The ID number is system-generated identification number for the circuit.
- The Circuit ID is created automatically, based on the service. For example, for Ethernet, it is based on the VLAN number; for Frame Relay, it is based on the DLCI; for ATM, it is based on the VPI/VCI.

**Step 18** Click **Save**.

The EWS (EPL) service request is created and saved in ISC.

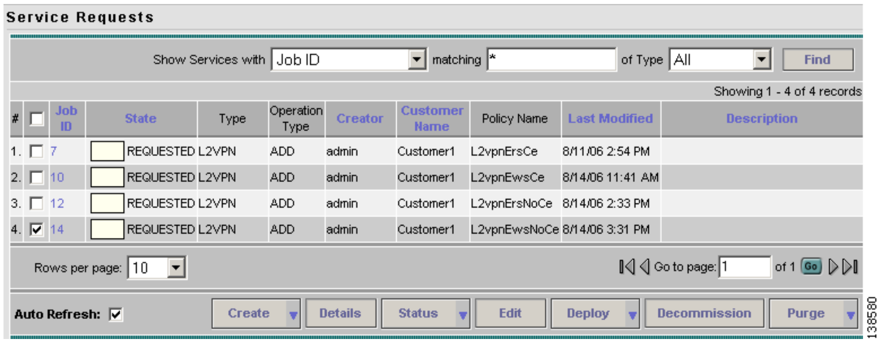
# Modifying the L2VPN Service Request

This section describes how to edit the L2VPN service request attributes. This is also where you can associate templates and data files to devices that are part of the ACs.

Perform the following steps.

**Step 1** Choose **Service Inventory** > **Inventory and Connection Manager** > **Service Requests**. (See Figure 8-30.)

*Figure 8-30 L2VPN Service Activation*



Service Requests

Show Services with Job ID ▾ matching * of Type All ▾ Find

Showing 1 - 4 of 4 records

| # | □ | Job ID | | State | Type | Operation Type | Creator | Customer Name | Policy Name | Last Modified | Description |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1. | □ | 7 | | REQUESTED | L2VPN | ADD | admin | Customer1 | L2vpnErsCe | 8/11/06 2:54 PM | |
| 2. | □ | 10 | | REQUESTED | L2VPN | ADD | admin | Customer1 | L2vpnEwsCe | 8/14/06 11:41 AM | |
| 3. | □ | 12 | | REQUESTED | L2VPN | ADD | admin | Customer1 | L2vpnErsNoCe | 8/14/06 2:33 PM | |
| 4. | ☑ | 14 | | REQUESTED | L2VPN | ADD | admin | Customer1 | L2vpnEwsNoCe | 8/14/06 3:31 PM | |

Rows per page: 10 ▾ Go to page: 1 of 1 Go

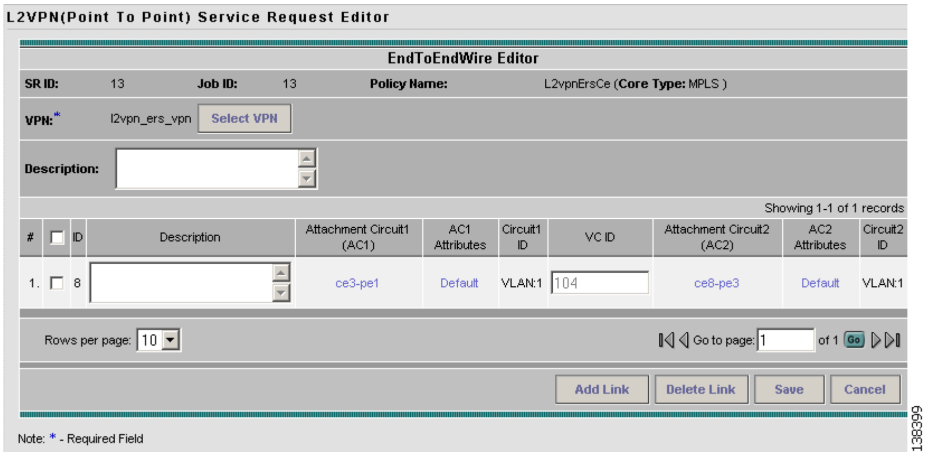
Auto Refresh: ☑

**Step 2** Check a check box for a service request.

**Step 3** Click **Edit**.

The End-to-End-Wire Editor window appears. (See Figure 8-31.)

*Figure 8-31 End-to-End Wire Editor*



L2VPN(Point To Point) Service Request Editor

EndToEndWire Editor

SR ID: 13 Job ID: 13 Policy Name: L2vpnErsCe (Core Type: MPLS )

VPN:* l2vpn_ers_vpn Select VPN

Description:

Showing 1-1 of 1 records

| # | □ | ID | Description | Attachment Circuit1 (AC1) | AC1 Attributes | Circuit1 ID | VC ID | Attachment Circuit2 (AC2) | AC2 Attributes | Circuit2 ID |
|---|---|---|---|---|---|---|---|---|---|---|
| 1. | □ | 8 | | ce3-pe1 | Default | VLAN:1 | 104 | ce8-pe3 | Default | VLAN:1 |

Rows per page: 10 ▾ Go to page: 1 of 1 Go

Note: * - Required Field

**Step 4** Modify any of the attributes, as desired:

- The VPN for this service request appears in the Select VPN field. If this request has more than one VPN, click **Select VPN** to choose a VPN.
- You can choose any of the blue highlighted values to edit the End-to-End Wire.
- You can edit the AC link attributes to change the default policy settings. After you edit these fields, the blue link changes from Default to Changed.
- You can enter a description for the service request in the first **Description** field. The description will show up in this window and also in the Description column of the Service Requests window. The maximum length for this field is 256 characters.
- You can enter a description for each end-to-end wire in the **Description** field provided for each wire. The description shows up only in this window. The data in this field is not pushed to the device(s). The maximum length for this field is 256 characters.
- The Circuit ID is created automatically, based on the VLAN data for the circuit.

- If the policy was set up for you to define a VC ID manually, enter it into the empty **VC ID** field. If policy was set to “auto pick” the VC ID, ISC will supply a VC ID, and this field will not be editable. In the case where you supply the VC ID manually, if the entered value is in the provider’s range, ISC validates if the entered value is available or allocated. If the entered value has been already allocated, ISC generates an error message saying that the entered value is not available and prompts you to re-enter the value. If the entered value is in the provider’s range, and if it is available, then it is allocated and is removed from the VC ID pool. If the entered value is outside the provider’s range, ISC displays a warning saying that no validation could be performed to verify if it is available or allocated.
- You can also click **Add Link** to add an end-to-end wire.
- You can click **Delete Link** to delete an end-to-end wire.

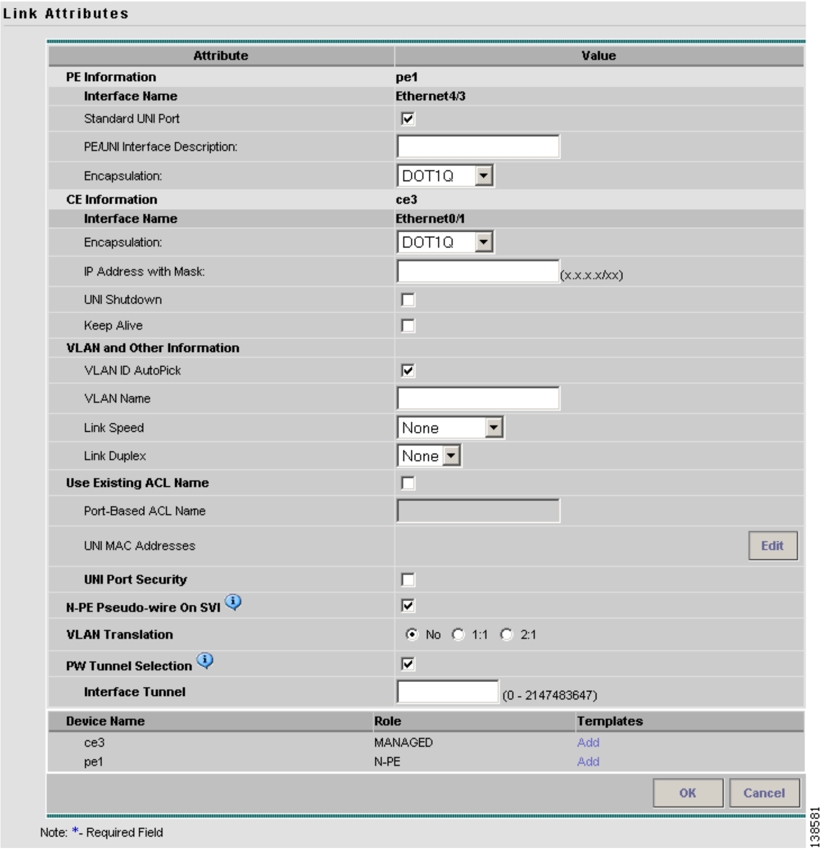
**Note** If you are attempting to decommission a service request to which a template has been added, see Monitoring Service Requests, page 11-10 for information on the proper way to do this.

- The ID number is system-generated identification number for the circuit.
- The Circuit ID is created automatically, based on the service. For example, for Ethernet, it is based on the VLAN number; for Frame Relay, it is based on the DLCI; for ATM, it is based on the VPI/VCI.

**Step 5** To edit AC attributes, click the **Default** button.

The Link Attributes window appears. (See Figure 8-32.)

*Figure 8-32 Link Attributes Window*



Link Attributes

| Attribute | Value |
|---|---|
| PE Information | pe1 |
| Interface Name | Ethernet4/3 |
| Standard UNI Port | ☑ |
| PE/UNI Interface Description: | |
| Encapsulation: | DOT1Q |
| CE Information | ce3 |
| Interface Name | Ethernet0/1 |
| Encapsulation: | DOT1Q |
| IP Address with Mask: | (x.x.x.x/xx) |
| UNI Shutdown | ☐ |
| Keep Alive | ☐ |
| VLAN and Other Information | |
| VLAN ID AutoPick | ☑ |
| VLAN Name | |
| Link Speed | None |
| Link Duplex | None |
| Use Existing ACL Name | ☐ |
| Port-Based ACL Name | |
| UNI MAC Addresses | Edit |
| UNI Port Security | ☐ |
| N-PE Pseudo-wire On SVI | ☑ |
| VLAN Translation | ● No ○ 1:1 ○ 2:1 |
| PW Tunnel Selection | ☑ |
| Interface Tunnel | (0 - 2147483647) |

| Device Name | Role | Templates |
|---|---|---|
| ce3 | MANAGED | Add |
| pe1 | N-PE | Add |

Note: *- Required Field

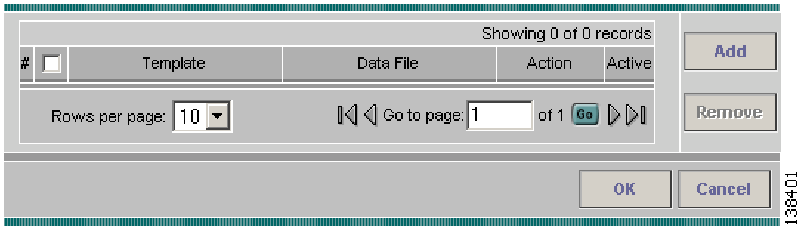
**Step 6** Edit any of the link attributes, as desired.

**Step 7** To add a template and data file to an attachment circuit, choose a Device Name, and click **Add** under Templates.

The Add/Remove Templates window appears. (See Figure 8-33.)

**Note** To add a template to an attachment circuit, you must have already created the template. For detailed steps to create templates, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*. For more information on how to use templates and data files in service requests, see Appendix B, "Working with Templates and Data Files."

*Figure 8-33 Add/Remove Templates*



Showing 0 of 0 records

| # | ☐ | Template | Data File | Action | Active |
|---|---|---|---|---|---|

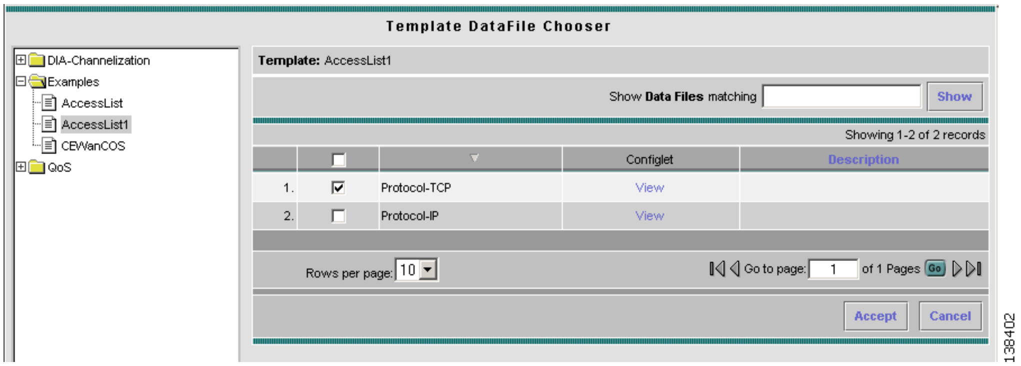
Rows per page: 10

Go to page: 1 of 1

**Step 8** Click **Add**.

The Template Data File Chooser window appears. (See Figure 8-34.)

*Figure 8-34 Template Datafile Chooser*



Template DataFile Chooser

- DIA-Channelization
- Examples
  - AccessList
  - AccessList1
  - CEWanCOS
- QoS

**Template:** AccessList1

Show **Data Files** matching [ ] Show

Showing 1-2 of 2 records

| | ☐ | | Configlet | Description |
|---|---|---|---|---|
| 1. | ☑ | Protocol-TCP | View | |
| 2. | ☐ | Protocol-IP | View | |

Rows per page: 10 Go to page: 1 of 1 Pages Go

Accept Cancel

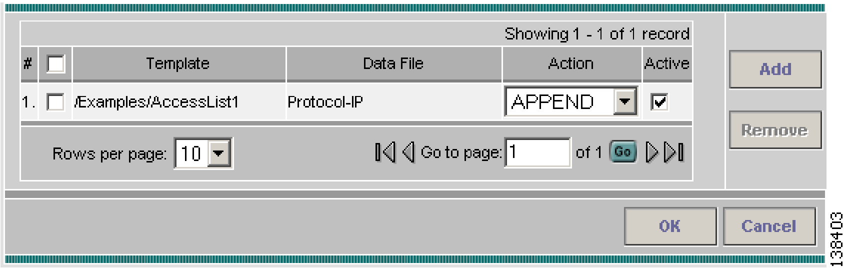
**Step 9** In the left pane, navigate to and select a template.

The associated data files are listed in rows in the main window, as shown in Figure 8-34.

**Step 10** Check the data file that you want to add and click **Accept**.

The Add/Remove Templates window appears with the template displayed. (See Figure 8-35.)

*Figure 8-35 Add/Remove Templates with Templates Shown*



Showing 1 - 1 of 1 record

| # | ☐ | Template | Data File | Action | Active |
|---|---|---|---|---|---|
| 1. | ☐ | /Examples/AccessList1 | Protocol-IP | APPEND | ☑ |

Rows per page: 10 Go to page: 1 of 1 Go

Add Remove

OK Cancel

**Step 11** Choose a Template name.

**Step 12** Under Action, use the drop-down list and choose **APPEND** or **PREPEND.**

Append tells ISC to append the template generated CLI to the regular ISC (non-template) CLI. Prepend is the reverse and does not append the template to the ISC CLI.

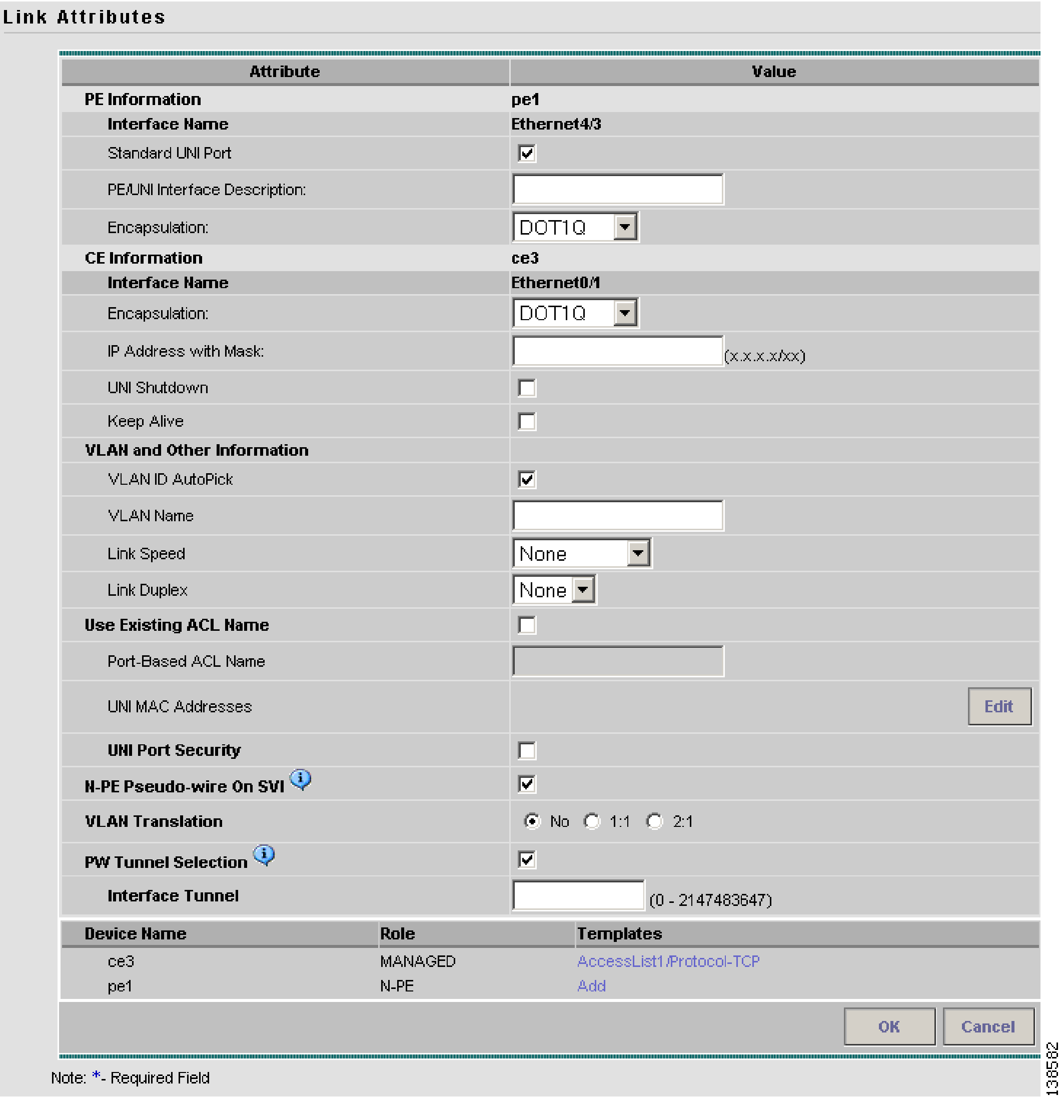
**Step 13** Choose **Active** to use this template for this service request.

If you do not choose Active, the template is not used.

**Step 14** Click **OK**.

The Link Attributes with the template added appears. (See Figure 8-36.)

*Figure 8-36 Link Attributes with Template Added*



Link Attributes

| Attribute | Value |
|---|---|
| PE Information | pe1 |
| Interface Name | Ethernet4/3 |
| Standard UNI Port | ☑ |
| PE/UNI Interface Description: | |
| Encapsulation: | DOT1Q |
| CE Information | ce3 |
| Interface Name | Ethernet0/1 |
| Encapsulation: | DOT1Q |
| IP Address with Mask: | (x.x.x.x/xx) |
| UNI Shutdown | □ |
| Keep Alive | □ |
| VLAN and Other Information | |
| VLAN ID AutoPick | ☑ |
| VLAN Name | |
| Link Speed | None |
| Link Duplex | None |
| Use Existing ACL Name | □ |
| Port-Based ACL Name | |
| UNI MAC Addresses | Edit |
| UNI Port Security | □ |
| N-PE Pseudo-wire On SVI | ☑ |
| VLAN Translation | ● No ○ 1:1 ○ 2:1 |
| PW Tunnel Selection | ☑ |
| Interface Tunnel | (0 - 2147483647) |

| Device Name | Role | Templates |
|---|---|---|
| ce3 | MANAGED | AccessList1/Protocol-TCP |
| pe1 | N-PE | Add |

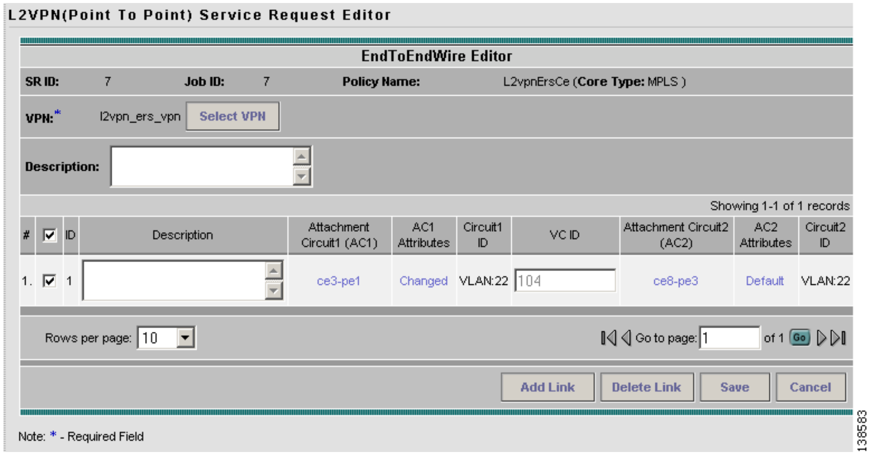
Note: *- Required Field

**Note** For more information about using templates and data files in service requests, see Appendix B, "Working with Templates and Data Files."

**Step 15** Click **OK**.

The Service Request Editor window appears showing the default for AC1 changed. (See Figure 8-37.)

*Figure 8-37 Service Request Editor with Link Attributes Changed.*



L2VPN(Point To Point) Service Request Editor

EndToEndWire Editor

SR ID: 7 Job ID: 7 Policy Name: L2vpnErsCe (Core Type: MPLS )

VPN:* l2vpn_ers_vpn Select VPN

Description:

Showing 1-1 of 1 records

| # | ☑ | ID | Description | Attachment Circuit1 (AC1) | AC1 Attributes | Circuit1 ID | VC ID | Attachment Circuit2 (AC2) | AC2 Attributes | Circuit2 ID |
|---|---|---|---|---|---|---|---|---|---|---|
| 1. | ☑ | 1 |  | ce3-pe1 | Changed | VLAN:22 | 104 | ce8-pe3 | Default | VLAN:22 |

Rows per page: 10 Go to page: 1 of 1 Go

Add Link Delete Link Save Cancel

Note: * - Required Field

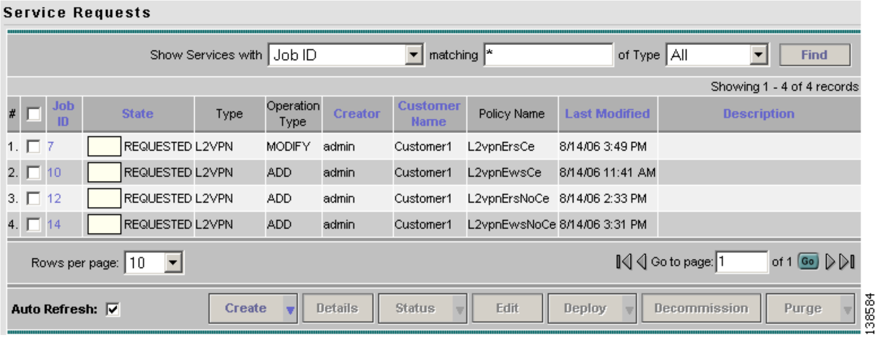
**Step 16** When you are finished editing the end-to-end wires, click **Save**.

# Saving the L2VPN Service Request

To save an L2VPN service request, perform the following steps.

**Step 1** When you are finished with Link Attributes for all the Attachment Circuits, click **Save** to finish the L2VPN service request creation. If the L2VPN service request is successfully created, you will see the service request list window. (See Figure 8-38.) The newly created L2VPN service request is added with the state of REQUESTED, as shown in the figure.

*Figure 8-38 L2VPN Service Request Created*



Service Requests

Show Services with Job ID matching * of Type All Find

Showing 1 - 4 of 4 records

| # | □ | Job ID | State | Type | Operation Type | Creator | Customer Name | Policy Name | Last Modified | Description |
|---|---|---|---|---|---|---|---|---|---|---|
| 1. | □ | 7 | REQUESTED | L2VPN | MODIFY | admin | Customer1 | L2vpnErsCe | 8/14/06 3:49 PM |  |
| 2. | □ | 10 | REQUESTED | L2VPN | ADD | admin | Customer1 | L2vpnEwsCe | 8/14/06 11:41 AM |  |
| 3. | □ | 12 | REQUESTED | L2VPN | ADD | admin | Customer1 | L2vpnErsNoCe | 8/14/06 2:33 PM |  |
| 4. | □ | 14 | REQUESTED | L2VPN | ADD | admin | Customer1 | L2vpnEwsNoCe | 8/14/06 3:31 PM |  |

Rows per page: 10 Go to page: 1 of 1 Go

Auto Refresh: ☑ Create Details Status Edit Deploy Decommission Purge

**Step 2** If, however, the L2VPN service request creation failed for some reason (for example, a value chosen is out of bounds), you are warned with an error message. In such a case, you should correct the error and save the service request again.

For information on deploying L2VPN service requests, see Deploying Service Requests, page 11-1.

CHAPTER 9

# Creating a VPLS Policy

This chapter contains the basic steps to create a VPLS policy. It contains the following sections:

## Defining a VPLS Policy

You must define a VPLS policy before you can provision a service. A VPLS policy defines the common characteristics shared by the Attachment Circuit (AC) attributes.

A policy can be shared by one or more service requests that have similar service requirements. The Editable check box gives the network operator the option of making a field editable. If the value is set to editable, the service request creator can change to other valid values for the particular policy item. If the value is *not* set to editable, the service request creator cannot change the policy item.

You can also associate Cisco IP Solution Center (ISC) templates and data files with a service request. See Appendix B, "Working with Templates and Data Files," for more about using templates and data files in service requests.

VPLS policies correspond to the one of the core types that VPLS provides:

- MPLS core type—provider core network is MPLS enabled
- Ethernet core type—provider core network uses Ethernet switches

and to one of the service types that VPLS provides:

- Ethernet Relay Multipoint Service (ERMS). The Metro Ethernet Forum name for ERMS is Ethernet Virtual Private LAN (EVP-LAN). See Layer 2 Terminology Conventions, page E-1 for more information about terms used to denote VPLS services in this guide.
- Ethernet Multipoint Service (EMS). The MEF name for EMS is Ethernet Private LAN (EP-LAN).

A policy is a template of most of the parameters needed to define a VPLS service request. After you define it, a VPLS policy can be used by all the VPLS service requests that share a common set of characteristics.

You create a new VPLS policy whenever you create a new type of service or a service with different parameters. VPLS policy creation is normally performed by experienced network engineers.

To define a VPLS policy in the Cisco IP Solution Center (ISC), perform the following steps.

**Step 1** Choose **Service Design** > **Policies**.

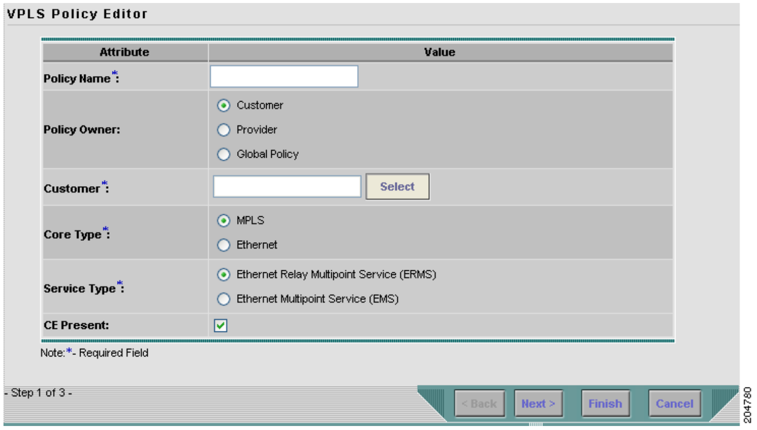
The Policies window appears.

**Step 2** Click **Create**.

**Step 3** Choose **VPLS Policy**.

The VPLS Policy Editor window appears. (See Figure 9-1.)

***Figure 9-1 Creating a VPLS Policy***



VPLS Policy Editor

| Attribute | Value |
|---|---|
| Policy Name*: | |
| Policy Owner: | ● Customer<br>○ Provider<br>○ Global Policy |
| Customer*: | [Select] |
| Core Type*: | ● MPLS<br>○ Ethernet |
| Service Type*: | ● Ethernet Relay Multipoint Service (ERMS)<br>○ Ethernet Multipoint Service (EMS) |
| CE Present: | ☑ |

Note:*- Required Field

- Step 1 of 3 -

**Step 4** Enter a **Policy Name** for the VPLS policy.

**Step 5** Choose the **Policy Owner** for the VPLS policy.

There are three types of VPLS policy ownership:

- Customer ownership
- Provider ownership
- Global ownership—Any service operator can make use of this VPLS policy.

This ownership has relevance when the ISC Role-Based Access Control (RBAC) comes into play. For example, a VPLS policy that is customer owned can only be seen by operators who are allowed to work on this customer-owned policy.

Similarly, operators who are allowed to work on a provider's network can view, use, and deploy a particular provider-owned policy.

**Step 6** Click **Select** to choose the owner of the VPLS policy.

The policy owner was established when you created customers or providers during ISC setup. If the ownership is global, the Select function does not appear.

**Step 7** Choose the **Core Type** of the VPLS policy.

There are two core types for VPLS policies:

- MPLS—running on an IP network
- Ethernet—all PEs are on an Ethernet provider network

**Step 8** Choose the **Service Type** of the VPLS policy.

There are two service types for VPLS policies:

- Ethernet Relay Multipoint Service (ERMS). (The MEF name for ERMS is EVP-LAN.)
- Ethernet Multipoint Service (EMS). (The MEF name for EMS is EP-LAN.)

**Step 9** Check the **CE Present** check box if you want ISC to ask the service operator who uses this VPLS policy to provide a CE router and interface during service activation.

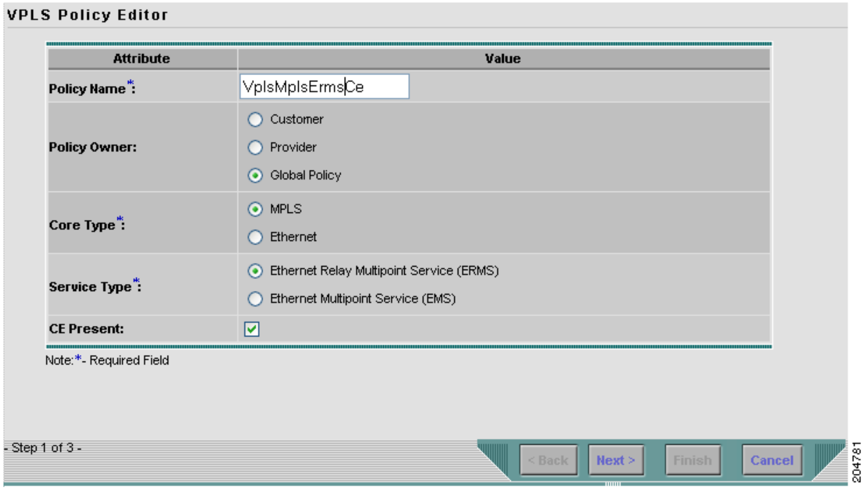
The default is CE present in the service.

If you do not check the **CE Present** check box, ISC asks the service operator, during service activation, only for the PE router and customer-facing interface.

# Defining an MPLS/ERMS (EVP-LAN) Policy with a CE

This section describes how to define a VPLS policy with an MPLS core type and an ERMS (EVP-LAN) service type with CE present. Figure 9-2 is an example of the first page of this policy.

*Figure 9-2 MPLS/ERMS (EVP-LAN) Policy with a CE*



VPLS Policy Editor

| Attribute | Value |
|---|---|
| Policy Name*: | VplsMplsErmsCe |
| Policy Owner: | ○ Customer<br>○ Provider<br>● Global Policy |
| Core Type*: | ● MPLS<br>○ Ethernet |
| Service Type*: | ● Ethernet Relay Multipoint Service (ERMS)<br>○ Ethernet Multipoint Service (EMS) |
| CE Present: | ☑ |

Note:*- Required Field

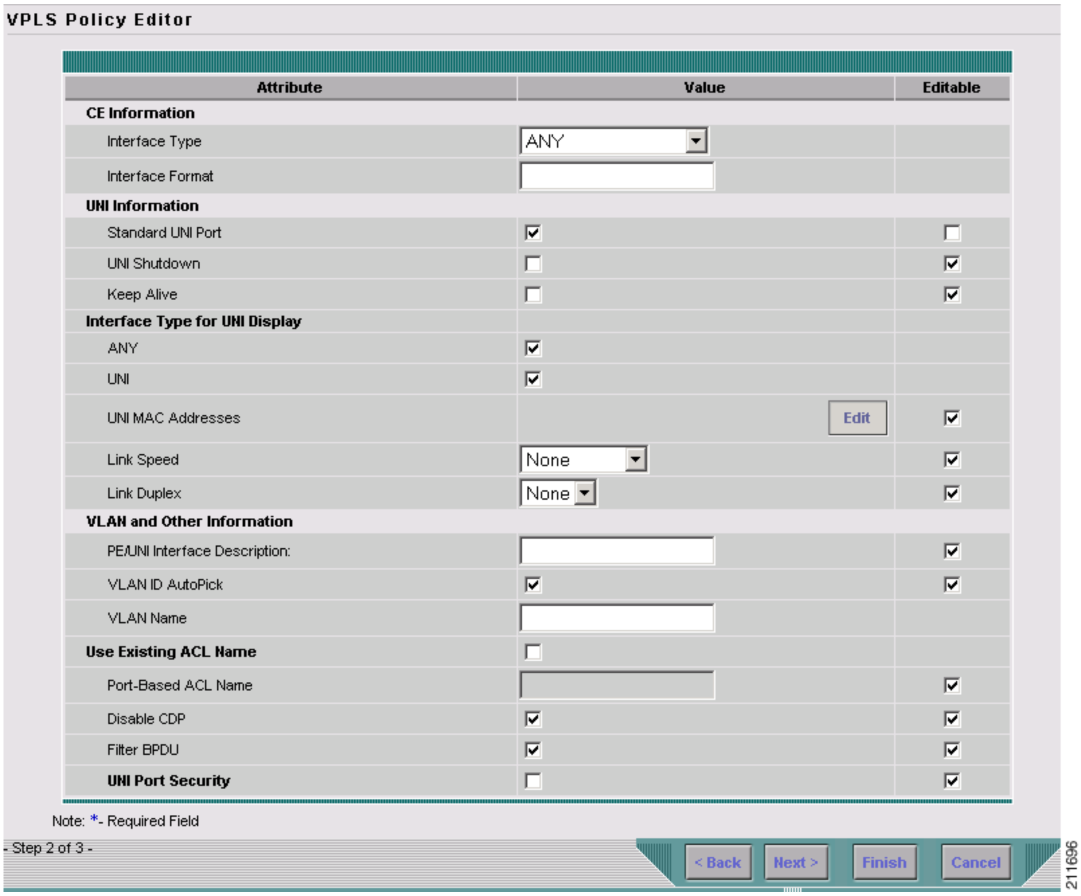
- Step 1 of 3 -

Perform the following steps.

**Step 1** Click **Next**. The window in Figure 9-3 appears.

The **Editable** check box gives you the option of making a field editable. If you check the **Editable** check box, the service operator who is using this VPLS policy can modify the editable parameter during VPLS service request creation.

*Figure 9-3 MPLS/ERMS (EVP-LAN) with a CE Policy Attributes*



VPLS Policy Editor

| Attribute | Value | Editable |
|---|---|---|
| **CE Information** | | |
| Interface Type | ANY | |
| Interface Format | | |
| **UNI Information** | | |
| Standard UNI Port | ☑ | ☐ |
| UNI Shutdown | ☐ | ☑ |
| Keep Alive | ☐ | ☑ |
| **Interface Type for UNI Display** | | |
| ANY | ☑ | |
| UNI | ☑ | |
| UNI MAC Addresses | Edit | ☑ |
| Link Speed | None | ☑ |
| Link Duplex | None | ☑ |
| **VLAN and Other Information** | | |
| PE/UNI Interface Description: | | ☑ |
| VLAN ID AutoPick | ☑ | ☑ |
| VLAN Name | | |
| **Use Existing ACL Name** | ☐ | |
| Port-Based ACL Name | | ☑ |
| Disable CDP | ☑ | ☑ |
| Filter BPDU | ☑ | ☑ |
| **UNI Port Security** | ☐ | ☑ |

Note: *- Required Field

- Step 2 of 3 -

**Step 2** Choose an **Interface Type** from the drop-down list.

You can choose a particular interface on a CE, N-PE, PE-AGG, or U-PE interface based on the service provider's POP design. The interfaces are:

- **ANY** (Any interface can be chosen.)
- **Port-Channel** (A bundle of ports that share the same characteristics—this gives the service provider the ability to aggregate bandwidth and protection.)
- **Ethernet**
- **FastEthernet**
- **GE-WAN**
- **GigabitEthernet**
- **TenGigabitEthernet**
- **TenGigE**

The value defined here functions as a filter to restrict the interface types an operator can see during VPLS service request creation. If defined as ANY, the operator can see all interface types.

**Step 3** Enter an **Interface Format** as the slot number/port number for the CE interface (for example, **1/0** indicates that the interface is located at slot 1, port 0).

This is especially useful to specify here if you know that the link will always go through a particular interface's slot/port location on all or most of the network devices in the service.

**Step 4** Choose a CE **Encapsulation** type.

The choices are:

- **DOT1Q**
- **DEFAULT**

If **DEFAULT** is the CE encapsulation type, ISC shows another field for the UNI port type.

**Step 5** Check the **Standard UNI Port** check box to enable port security.

This is the default. When you uncheck the check box, the port is treated as an uplink with no security features, and the window dynamically changes to eliminate items related to port security.

**Step 6** Check **UNI Shutdown** box if you want to leave the UNI port shut during service activation, for example, when the service provider wants to deploy a service in the network but wants to activate it at a later time.

**Step 7** Check the **Keep Alive** check box to configure keepalives on the UNI port.

By default, this check box is unchecked, which causes the command **no keepalive** to be provisioned on the UNI port. This prevents a CPE from sending keepalive packets to the U-PE, for security purposes. This attribute is editable to support modification on a per-service request basis.

**Step 8** Check the **ANY** check box to display all interface types as choices for the UNI interface (when creating service requests based on this policy).

This check box is checked by default.

**Step 9** Check the **UNI** check box to display all interfaces defined as type UNI as choices for the UNI interface (when creating service requests based on this policy).

This check box is checked by default.

**Step 10** Enter one or more Ethernet MAC addresses in **UNI MAC addresses**.

This selection is present only if you uncheck the **Use Existing ACL Name** check box. Click the **Edit** button to bring up a pop-up window in which you enter MAC addresses to be allowed or denied on the port. You can also specify a range of addresses by setting a base MAC address and a filtered MAC address.

**Step 11** Choose a **Port Type**.

The choices are:

- **Access Port**
- **Trunk with Native VLAN**

**Step 12** Enter a **Link Speed** (optional) of None, 10, 100, 1000, Auto, or nonegotiate.

**Step 13** Enter a **Link Duplex** (optional) of None, Full, Half, or Auto.

**Step 14** In the **PE/UNI Interface Description** field, enter an optional description, for example *Customer-B ERMS (EVP-LAN) Service*.

**Step 15** Check the **VLAN ID AutoPick** check box if you want ISC to choose a VLAN ID.

If you do not check this check box, you will be prompted to provide the VLAN in a Provider VLAN ID field during service activation.

**Step 16** Enter a **VLAN NAME** (optional) to specify a name to describe the VLAN.

The name must be one token (no spaces allowed.) The limit for the VLAN name is 32 characters. The name has to be unique. Two VLANs cannot share the same name.

**Step 17** Check the **Use Existing ACL Name** check box if you want assign your own named access list to the port.

By default, this check box is not checked and ISC automatically assigns a MAC-based ACL on the customer facing UNI port, based on values you enter in **UNI MAC addresses** (below).

**Step 18** Enter a **Port-Based ACL Name** (if you checked the **Use Existing ACL Name** check box, as mentioned in the previous step).

**Note** ISC does not create this ACL automatically. The ACL must already exist on the device, or be added as part of a template, before the service request is deployed. Otherwise, deployment will fail.

**Step 19** Check the **Disable CDP** check box if you want to disable the Cisco Discover Protocol (CDP) on the UNI port.

**Step 20** Check the **Filter BPDU** check box to specify that the UNI port should not process Layer 2 Bridge Protocol Data Units (BPDUs).

**Step 21** Check the **UNI Port Security** check box (see Figure 9-4) if you to want to provision port security-related CLIs to the UNI port by controlling the MAC addresses that are allowed to go through the interface.

**a.** For **Maximum Number of MAC address**, enter the number of MAC addresses allowed for port security.

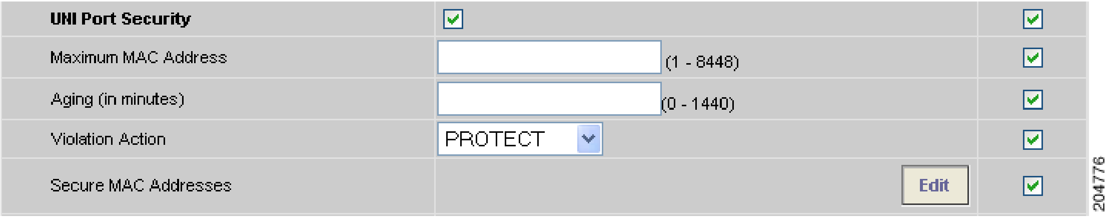
**b.** For **Aging,** enter the length of time the MAC address can stay on the port security table.

**c.** For **Violation Action**, choose what action will occur when a port security violation is detected:

- **PROTECT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value.
- **RESTRICT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value and causes the Security Violation counter to increment.
- **SHUTDOWN**—Puts the interface into the error-disabled state immediately and sends an SNMP trap notification.

**d.** In the **Secure MAC Addresses** field, enter one or more Ethernet MAC addresses. Click the **Edit** button to enter the addresses.

***Figure 9-4 UNI Port Security***

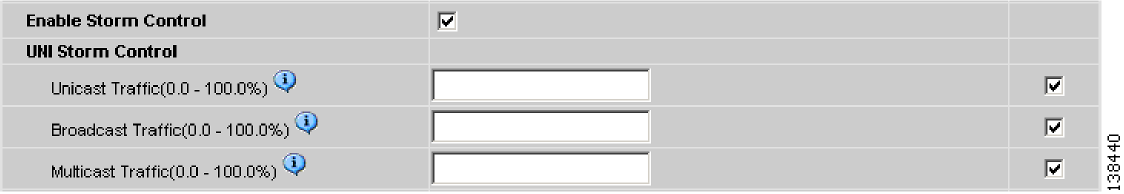


| | | |
|---|---|---|
| **UNI Port Security** | ☑ | ☑ |
| Maximum MAC Address | (1 - 8448) | ☑ |
| Aging (in minutes) | (0 - 1440) | ☑ |
| Violation Action | PROTECT | ☑ |
| Secure MAC Addresses | Edit | ☑ |

**Step 22** Check the **Enable Storm Control** check box (see Figure 9-5) to help prevent the UNI port from being disrupted by a broadcast, multicast or unicast storm.

Enter a threshold value for each type of traffic. The value, which can be specified to two significant digits, represents the percentage of the total available bandwidth of the port. If the threshold of a traffic type is reached, further traffic of that type is suppressed until the incoming traffic falls below the threshold level.

*Figure 9-5 Enable Storm Control*



| | | |
|---|---|---|
| **Enable Storm Control** | ☑ | |
| **UNI Storm Control** | | |
| Unicast Traffic(0.0 - 100.0%) | | ☑ |
| Broadcast Traffic(0.0 - 100.0%) | | ☑ |
| Multicast Traffic(0.0 - 100.0%) | | ☑ |

**Step 23** Click the **Next** button, if you want to enable template support for the policy.

The Template Association window appears. In this window, you can enable template support and, optionally, associate templates and data files with the policy. For instructions about associating templates with policies and how to use the features in this window, see Appendix B, "Working with Templates and Data Files". When you have completed setting up templates and data files for the policy, click **Finish** in the Template Association window to close it and return to the Policy Editor window.

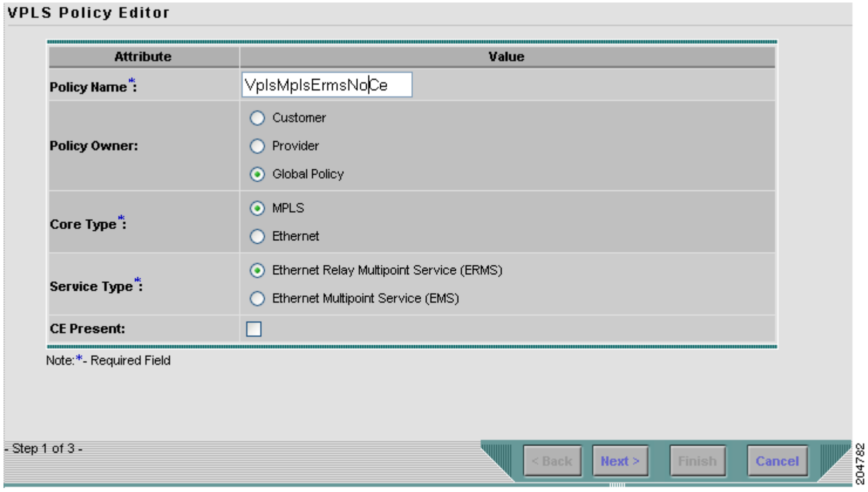
**Step 24** Click **Finish**.

**Note** The VC ID is mapped from the VPN ID. By default, ISC will "auto pick" this value. However, you can set this manually, if desired. This is done by editing the associated VPN configuration. The Edit VPN window has an **Enable VPLS** check box. When you check this box, you can manually enter a VPN ID in a field provided. For more information on creating and modifying VPNs, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*.

# Defining an MPLS/ERMS (EVP-LAN) Policy without a CE

This section describes defining a VPLS policy with an MPLS core type and an ERMS (EVP-LAN) service type without a CE present. Figure 9-6 is an example of the first page of this policy.

*Figure 9-6 MPLS/ERMS (EVP-LAN) Policy without a CE*



VPLS Policy Editor

| Attribute | Value |
|---|---|
| Policy Name*: | VplsMplsErmsNoCe |
| Policy Owner: | ○ Customer<br>○ Provider<br>● Global Policy |
| Core Type*: | ● MPLS<br>○ Ethernet |
| Service Type*: | ● Ethernet Relay Multipoint Service (ERMS)<br>○ Ethernet Multipoint Service (EMS) |
| CE Present: | □ |

Note:*- Required Field

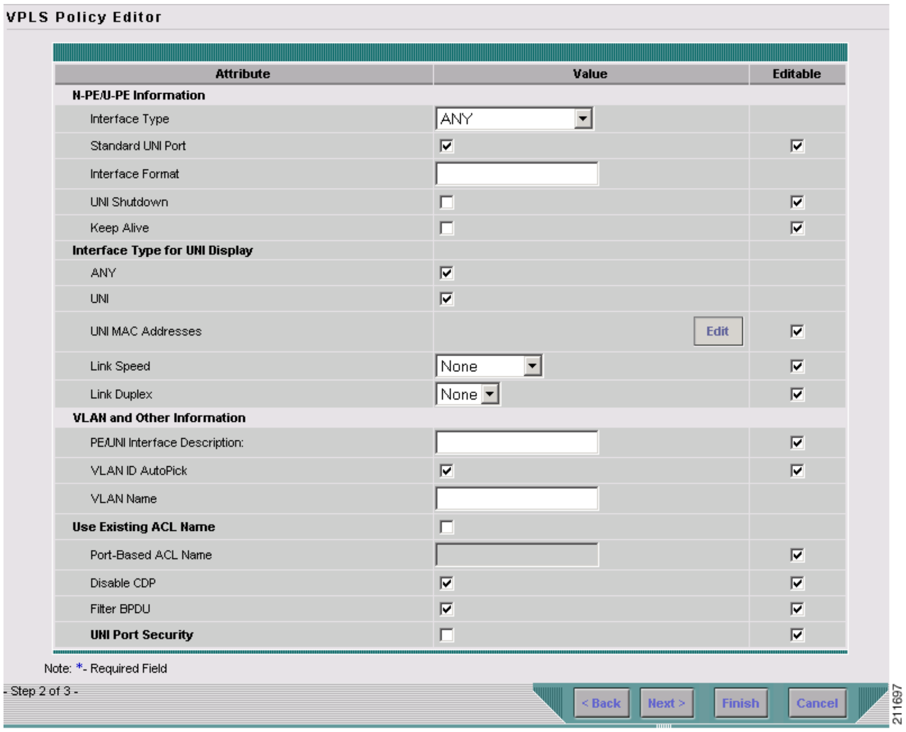
- Step 1 of 3 -

Perform the following steps.

**Step 1** Click **Next**. The window in Figure 9-7 appears.

The **Editable** check box gives you the option of making a field editable. If you check the **Editable** check box, the service operator who is using this VPLS policy can modify the editable parameter during VPLS service request creation.

*Figure 9-7 MPLS/ERMS (EVP-LAN) without a CE Policy Attributes*



VPLS Policy Editor

| Attribute | Value | Editable |
|---|---|---|
| **N-PE/U-PE Information** | | |
| Interface Type | ANY ▾ | |
| Standard UNI Port | ☑ | ☑ |
| Interface Format | | |
| UNI Shutdown | ☐ | ☑ |
| Keep Alive | ☐ | ☑ |
| **Interface Type for UNI Display** | | |
| ANY | ☑ | |
| UNI | ☑ | |
| UNI MAC Addresses | Edit | ☑ |
| Link Speed | None ▾ | ☑ |
| Link Duplex | None ▾ | ☑ |
| **VLAN and Other Information** | | |
| PE/UNI Interface Description: | | ☑ |
| VLAN ID AutoPick | ☑ | ☑ |
| VLAN Name | | |
| **Use Existing ACL Name** | ☐ | |
| Port-Based ACL Name | | ☑ |
| Disable CDP | ☑ | ☑ |
| Filter BPDU | ☑ | ☑ |
| **UNI Port Security** | ☐ | ☑ |

Note: *- Required Field

- Step 2 of 3 -

**Step 2** Choose an **Interface Type** from the drop-down list.

You can choose a particular interface on a N-PE, U-PE, or PE-AGG interface based on the service provider's POP design. The interfaces are:

- **ANY** (Any interface can be chosen.)
- **Port-Channel** (A bundle of ports that share the same characteristics—this gives the service provider the ability to aggregate bandwidth and protection.)
- **Ethernet**
- **FastEthernet**
- **GE-WAN**
- **GigabitEthernet**
- **TenGigabitEthernet**
- **TenGigE**

The value defined here functions as a filter to restrict the interface types an operator can see during VPLS service request creation. If defined as ANY, the operator can see all interface types.

**Step 3** Check the **Standard UNI Port** check box to enable port security.

This is the default. When you uncheck the check box, the port is treated as an uplink with no security features, and the window dynamically changes to eliminate items related to port security.

**Step 4** Enter an **Interface Format** as the slot number/port number for the CE interface (for example, **1/0** indicates that the interface is located at slot 1, port 0).

This is especially useful to specify here if you know that the link will always go through a particular interface’s slot/port location on all or most of the network devices in the service.

**Step 5** Choose a CE **Encapsulation** type.

The choices are:

- **DOT1Q**
- **DEFAULT**

If **DEFAULT** is the CE encapsulation type, ISC shows another field for the UNI port type.

**Step 6** Check **UNI Shutdown** box if you want to leave the UNI port shut during service activation, for example, when the service provider wants to deploy a service in the network but wants to activate it at a later time.

**Step 7** Check the **Keep Alive** check box to configure keepalives on the UNI port.

By default, this check box is unchecked, which causes the command **no keepalive** to be provisioned on the UNI port. This prevents a CPE from sending keepalive packets to the U-PE, for security purposes. This attribute is editable to support modification on a per-service request basis.

**Step 8** Check the **ANY** check box to display all interface types as choices for the UNI interface (when creating service requests based on this policy).

This check box is checked by default.

**Step 9** Check the **UNI** check box to display all interfaces defined as type UNI as choices for the UNI interface (when creating service requests based on this policy). This check box is checked by default.

**Step 10** Enter one or more Ethernet MAC addresses in **UNI MAC addresses**.

This selection is present only if you uncheck the **Use Existing ACL Name** check box. Click the **Edit** button to bring up a pop-up window in which you enter MAC addresses to be allowed or denied on the port. You can also specify a range of addresses by setting a base MAC address and a filtered MAC address.

**Step 11** Choose a **Port Type**.

The choices are:

- **Access Port**
- **Trunk with Native VLAN**

**Step 12** Enter a **Link Speed** (optional) of None, 10, 100, 1000, Auto, or nonegotiate.

**Step 13** Enter a **Link Duplex** (optional) of None, Full, Half, or Auto.

**Step 14** In the **PE/UNI Interface Description** field, enter an optional description, for example *Customer-B ERMS (EVP-LAN) Service*.

**Step 15** Check the **VLAN ID AutoPick** check box if you want ISC to choose a VLAN ID.

If you do not check this check box, you will be prompted to provide the VLAN in a Provider VLAN ID field during service activation.

**Step 16** Enter a **VLAN NAME** (optional) to specify a name to describe the VLAN.

The name must be one token (no spaces allowed.) The limit for the VLAN name is 32 characters. The name has to be unique. Two VLANs cannot share the same name.

**Step 17** Check the **Use Existing ACL Name** check box if you want assign your own named access list to the port.

By default, this check box is not checked and ISC automatically assigns a MAC-based ACL on the customer facing UNI port, based on values you enter in **UNI MAC addresses** (below).

**Step 18** Enter a **Port-Based ACL Name** (if you checked the **Use Existing ACL Name** check box, as mentioned in the previous step).

> **Note** ISC does not create this ACL automatically. The ACL must already exist on the device, or be added as part of a template, before the service request is deployed. Otherwise, deployment will fail.

**Step 19** Check the **Disable CDP** check box if you want to disable the Cisco Discover Protocol (CDP) on the UNI port.

**Step 20** Check the **Filter BPDU** check box to specify that the UNI port should not process Layer 2 Bridge Protocol Data Units (BPDUs).

**Step 21** Check the **UNI Port Security** check box (see Figure 9-8) if you to want to provision port security-related CLIs to the UNI port by controlling the MAC addresses that are allowed to go through the interface.

a. For **Maximum Number of MAC address**, enter the number of MAC addresses allowed for port security.

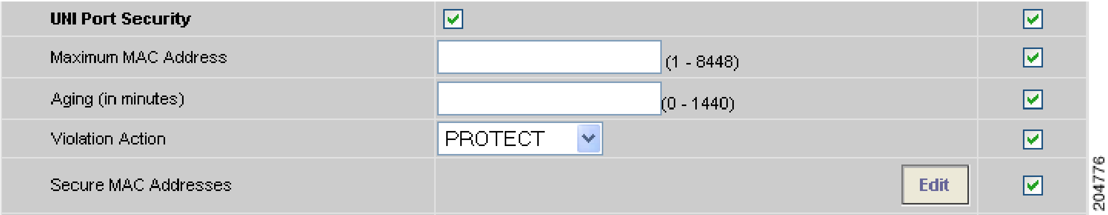
b. For **Aging,** enter the length of time the MAC address can stay on the port security table.

c. For **Violation Action**, choose what action will occur when a port security violation is detected:

- **PROTECT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value.
- **RESTRICT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value and causes the Security Violation counter to increment.
- **SHUTDOWN**—Puts the interface into the error-disabled state immediately and sends an SNMP trap notification.

d. In the **Secure MAC Addresses** field, enter one or more Ethernet MAC addresses.

*Figure 9-8 UNI Port Security*

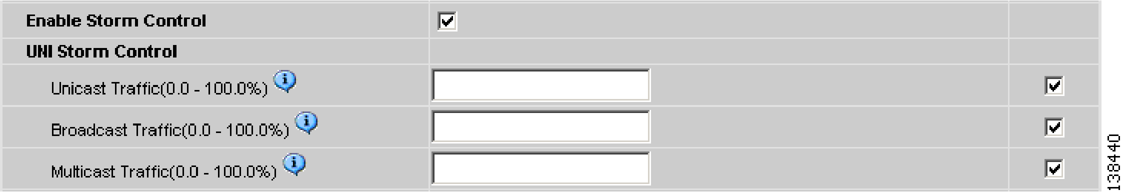


| | | |
|---|---|---|
| **UNI Port Security** | ☑ | ☑ |
| Maximum MAC Address | (1 - 8448) | ☑ |
| Aging (in minutes) | (0 - 1440) | ☑ |
| Violation Action | PROTECT | ☑ |
| Secure MAC Addresses | Edit | ☑ |

**Step 22** Check the **Enable Storm Control** check box (see Figure 9-9) to help prevent the UNI port from being disrupted by a broadcast, multicast or unicast storm.

Enter a threshold value for each type of traffic. The value, which can be specified to two significant digits, represents the percentage of the total available bandwidth of the port. If the threshold of a traffic type is reached, further traffic of that type is suppressed until the incoming traffic falls below the threshold level.

*Figure 9-9 Enable Storm Control*



| | | |
|---|---|---|
| **Enable Storm Control** | ☑ | |
| **UNI Storm Control** | | |
| Unicast Traffic(0.0 - 100.0%) | | ☑ |
| Broadcast Traffic(0.0 - 100.0%) | | ☑ |
| Multicast Traffic(0.0 - 100.0%) | | ☑ |

**Step 23** Click the **Next** button, if you want to enable template support for the policy.

The Template Association window appears. In this window, you can enable template support and, optionally, associate templates and data files with the policy. For instructions about associating templates with policies and how to use the features in this window, see Appendix B, "Working with Templates and Data Files". When you have completed setting up templates and data files for the policy, click **Finish** in the Template Association window to close it and return to the Policy Editor window.

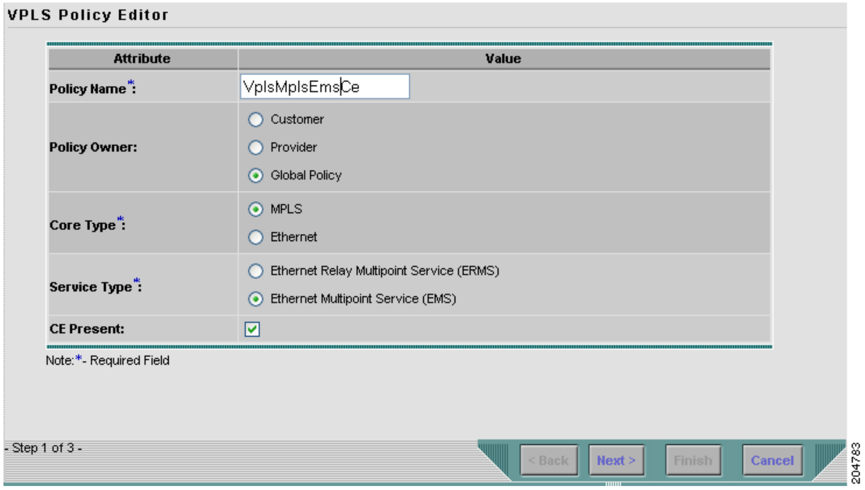
**Step 24** Click **Finish**.

**Note** The VC ID is mapped from the VPN ID. By default, ISC will "auto pick" this value. However, you can set this manually, if desired. This is done by editing the associated VPN configuration. The Edit VPN window has an **Enable VPLS** check box. When you check this box, you can manually enter a VPN ID in a field provided. For more information on creating and modifying VPNs, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*.

# Defining an MPLS/EMS (EP-LAN) Policy with a CE

This section describes defining a VPLS policy with an MPLS core type and an EMS (EP-LAN) service type with CE present. Figure 9-10 is an example of the first page of this policy.

*Figure 9-10 MPLS/EMS (EP-LAN) Policy with a CE*



VPLS Policy Editor

| Attribute | Value |
|---|---|
| Policy Name*: | VplsMplsEmsCe |
| Policy Owner: | ○ Customer<br>○ Provider<br>● Global Policy |
| Core Type*: | ● MPLS<br>○ Ethernet |
| Service Type*: | ○ Ethernet Relay Multipoint Service (ERMS)<br>● Ethernet Multipoint Service (EMS) |
| CE Present: | ☑ |

Note:*- Required Field

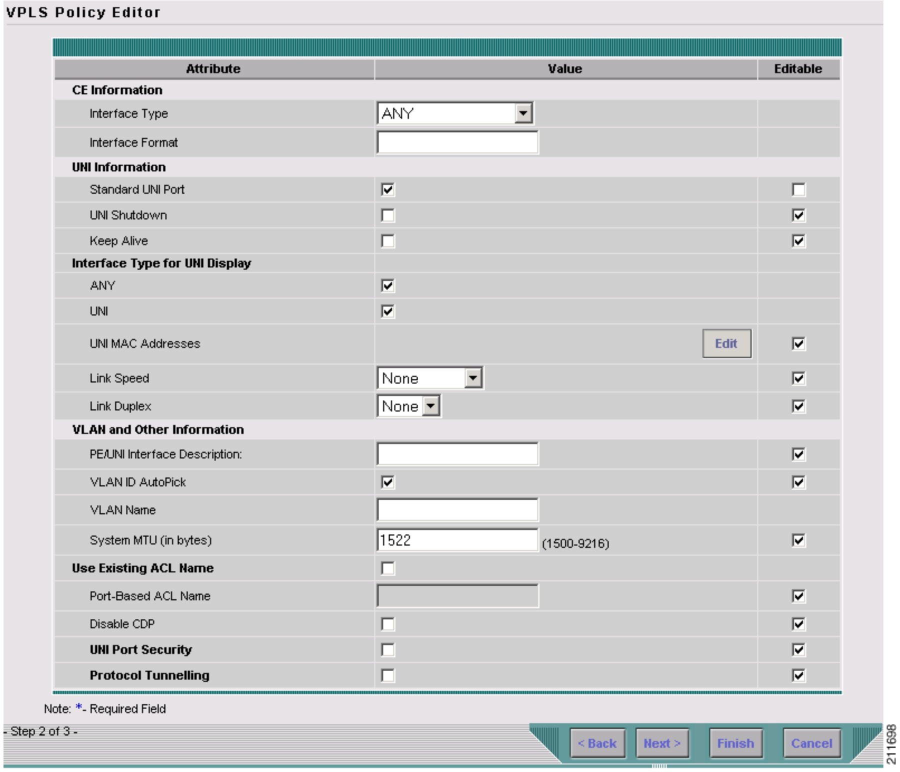
- Step 1 of 3 -

Perform the following steps.

**Step 1** Click **Next**. The window in Figure 9-11 appears.

The **Editable** check box gives you the option of making a field editable. If you check the **Editable** check box, the service operator who is using this VPLS policy can modify the editable parameter during VPLS service request creation.

*Figure 9-11 MPLS/EMS (EP-LAN) with a CE Policy Attributes*



VPLS Policy Editor

| Attribute | Value | Editable |
|---|---|---|
| **CE Information** | | |
| Interface Type | ANY | |
| Interface Format | | |
| **UNI Information** | | |
| Standard UNI Port | ☑ | ☐ |
| UNI Shutdown | ☐ | ☑ |
| Keep Alive | ☐ | ☑ |
| **Interface Type for UNI Display** | | |
| ANY | ☑ | |
| UNI | ☑ | |
| UNI MAC Addresses | Edit | ☑ |
| Link Speed | None | ☑ |
| Link Duplex | None | ☑ |
| **VLAN and Other Information** | | |
| PE/UNI Interface Description: | | ☑ |
| VLAN ID AutoPick | ☑ | ☑ |
| VLAN Name | | |
| System MTU (in bytes) | 1522 (1500-9216) | ☑ |
| **Use Existing ACL Name** | ☐ | |
| Port-Based ACL Name | | ☑ |
| Disable CDP | ☐ | ☑ |
| **UNI Port Security** | ☐ | ☑ |
| **Protocol Tunnelling** | ☐ | ☑ |

Note: *- Required Field

- Step 2 of 3 -

**Step 2** Choose an **Interface Type** from the drop-down list.

You can choose a particular interface on a CE, N-PE, U-PE, or PE-AGG interface based on the service provider's POP design. The interfaces are:

- **ANY** (Any interface can be chosen.)
- **Port-Channel** (A bundle of ports that share the same characteristics—this gives the service provider the ability to aggregate bandwidth and protection.)
- **Ethernet**
- **FastEthernet**
- **GE-WAN**
- **GigabitEthernet**
- **TenGigabitEthernet**
- **TenGigE**

The value defined here functions as a filter to restrict the interface types an operator can see during VPLS service request creation. If defined as ANY, the operator can see all interface types.

**Step 3** Enter an **Interface Format** as the slot number/port number for the CE interface (for example, **1/0** indicates that the interface is located at slot 1, port 0).

This is especially useful to specify here if you know that the link will always go through a particular interface's slot/port location on all or most of the network devices in the service.

**Step 4** Choose a CE **Encapsulation** type.

The choices are:

- **DOT1Q**
- **DEFAULT**

**Note** When creating a service request based on the MPLS/EMS (EP-LAN) with CE policy, the Encapsulation attribute is ignored. Therefore, setting this value has no effect.

**Step 5** Check the **Standard UNI Port** check box to enable port security.

This is the default. When you uncheck the check box, the port is treated as an uplink with no security features, and the window dynamically changes to eliminate items related to port security.

**Step 6** Check the **UNI Shutdown** check box if you want to leave the UNI port shut during service activation, for example, when the service provider wants to deploy a service in the network but wants to activate it at a later time.

**Step 7** Check the **Keep Alive** check box to configure keepalives on the UNI port.

By default, this check box is unchecked, which causes the command **no keepalive** to be provisioned on the UNI port. This prevents a CPE from sending keepalive packets to the U-PE, for security purposes. This attribute is editable to support modification on a per-service request basis.

**Step 8** Check the **ANY** check box to display all interface types as choices for the UNI interface (when creating service requests based on this policy). This check box is checked by default.

**Step 9** Enter one or more Ethernet MAC addresses in **UNI MAC addresses**.

This selection is present only if you uncheck the **Use Existing ACL Name** check box. Click the **Edit** button to bring up a pop-up window in which you enter MAC addresses to be allowed or denied on the port. You can also specify a range of addresses by setting a base MAC address and a filtered MAC address.

**Step 10** Check the **UNI** check box to display all interfaces defined as type UNI as choices for the UNI interface (when creating service requests based on this policy).

This check box is checked by default.

**Step 11** Enter a **Link Speed** (optional) of None, 10, 100, 1000, Auto, or nonegotiate.

**Step 12** Enter a **Link Duplex** (optional) of None, Full, Half, or Auto.

**Step 13** In the **PE/UNI Interface Description** field, enter an optional description, for example *Customer-B EMS (EP-LAN) Service*.

**Step 14** Check the **VLAN ID AutoPick** check box if you want ISC to choose a VLAN ID.

If you do not check this check box, you will be prompted to provide the VLAN in a Provider VLAN ID field during service activation.

**Step 15** Enter a **VLAN NAME** (optional) to specify a name to describe the VLAN.

The name must be one token (no spaces allowed.) The limit for the VLAN name is 32 characters. The name has to be unique. Two VLANs cannot share the same name.

**Step 16** Enter the **System MTU** in bytes.

The maximum transmission unit (MTU) size is configurable and optional. ISC does not perform an integrity check for this customized value. If a service request goes to the **Failed Deploy** state because this size is not accepted, you must adjust the size until the service request is deployed. ISC supports, ranges for different platforms, as specified below. The range is 1500 to 9216.

- For the 3750 and 3550 platforms, the MTU range is 1500-1546.
- For the 7600 ethernet port, the MTU size is always 9216. Even with the same platform and same IOS release, different line cards support the MTU differently. For example, older line cards only take an MTU size of 9216 and newer cards support 1500-9216. However, ISC uses 9216 in both cases.
- For the 7600 SVI (interface VLAN), the MTU size is 1500-9216.

**Step 17** Check the **Use Existing ACL Name** check box if you want assign your own named access list to the port.

By default, this check box is not checked and ISC automatically assigns a MAC-based ACL on the customer facing UNI port, based on values you enter in **UNI MAC addresses** (below).

**Step 18** Enter a **Port-Based ACL Name** (if you checked the **Use Existing ACL Name** check box, as mentioned in the previous step).

**Note** ISC does not create this ACL automatically. The ACL must already exist on the device, or be added as part of a template, before the service request is deployed. Otherwise, deployment will fail.

**Step 19** Check the **Disable CDP** check box if you want to disable the Cisco Discover Protocol (CDP) on the UNI port.

**Step 20** Check the **UNI Port Security** check box (see Figure 9-12) if you to want to provision port security-related CLIs to the UNI port by controlling the MAC addresses that are allowed to go through the interface.

a. For **Maximum Number of MAC address**, enter the number of MAC addresses allowed for port security.

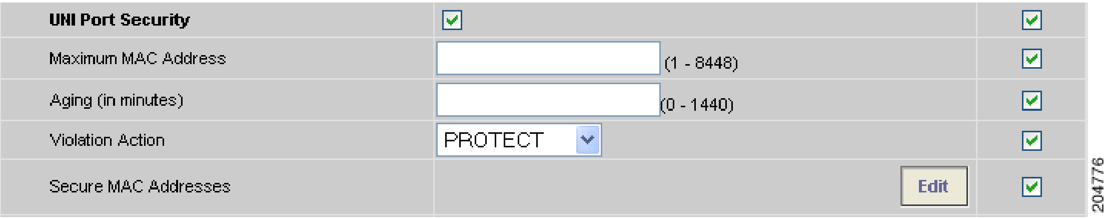
b. For **Aging,** enter the length of time the MAC address can stay on the port security table.

c. For **Violation Action**, choose what action will occur when a port security violation is detected:

- **PROTECT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value.
- **RESTRICT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value and causes the Security Violation counter to increment.
- **SHUTDOWN**—Puts the interface into the error-disabled state immediately and sends an SNMP trap notification.

d. In the **Secure MAC Addresses** field, enter one or more Ethernet MAC addresses.

*Figure 9-12 UNI Port Security*

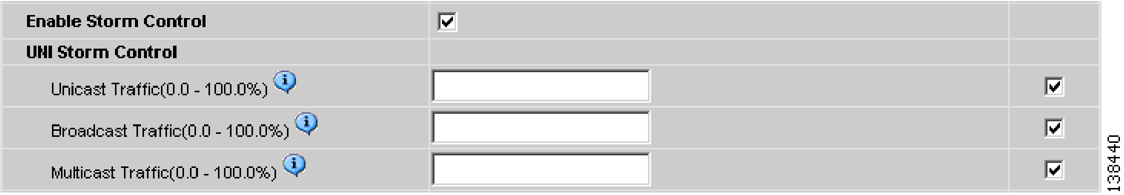


| | | | |
|---|---|---|---|
| **UNI Port Security** | ☑ | | ☑ |
| Maximum MAC Address | | (1 - 8448) | ☑ |
| Aging (in minutes) | | (0 - 1440) | ☑ |
| Violation Action | PROTECT | | ☑ |
| Secure MAC Addresses | | Edit | ☑ |

**Step 21** Check the **Enable Storm Control** check box (see Figure 9-13) to help prevent the UNI port from being disrupted by a broadcast, multicast or unicast storm.

Enter a threshold value for each type of traffic. The value, which can be specified to two significant digits, represents the percentage of the total available bandwidth of the port. If the threshold of a traffic type is reached, further traffic of that type is suppressed until the incoming traffic falls below the threshold level.

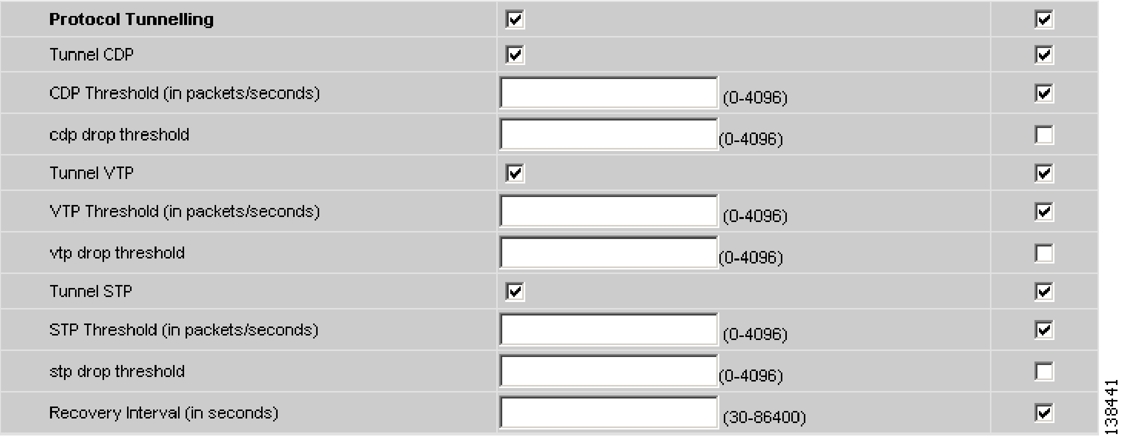
*Figure 9-13 Enable Storm Control*



| | | |
|---|---|---|
| **Enable Storm Control** | ☑ | |
| **UNI Storm Control** | | |
| Unicast Traffic(0.0 - 100.0%) | | ☑ |
| Broadcast Traffic(0.0 - 100.0%) | | ☑ |
| Multicast Traffic(0.0 - 100.0%) | | ☑ |

**Step 22** Check the **Protocol Tunnelling** check box (see Figure 9-14) if you want to define the Layer 2 Bridge Protocol Data Unit (BPDU) frames that can be tunneled over the core to the other end.

*Figure 9-14 Protocol Tunnelling*



| | | | |
|---|---|---|---|
| **Protocol Tunnelling** | ☑ | | ☑ |
| Tunnel CDP | ☑ | | ☑ |
| CDP Threshold (in packets/seconds) | | (0-4096) | ☑ |
| cdp drop threshold | | (0-4096) | ☐ |
| Tunnel VTP | ☑ | | ☑ |
| VTP Threshold (in packets/seconds) | | (0-4096) | ☑ |
| vtp drop threshold | | (0-4096) | ☐ |
| Tunnel STP | ☑ | | ☑ |
| STP Threshold (in packets/seconds) | | (0-4096) | ☑ |
| stp drop threshold | | (0-4096) | ☐ |
| Recovery Interval (in seconds) | | (30-86400) | ☑ |

For each protocol that you check, enter the shutdown threshold and drop threshold for that protocol:

a. **Tunnel CDP**—Enable Layer 2 tunnelling on Cisco Discover Protocol (CDP).

b. **CDP Threshold**—Enter the number of packets per second to be received before the interface is shut down.

c. **cdp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping CDP packets.

d. **Tunnel VTP**—Enable Layer 2 tunnelling on VLAN Trunk Protocol (VTP).

e. **VTP threshold**—Enter the number of packets per second to be received before the interface is shut down.

f. **vtp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping VTP packets.

g. **Tunnel STP**—Enable Layer 2 tunnelling on Spanning Tree Protocol (STP).

h. **STP Threshold**—Enter the number of packets per second to be received before the interface is shut down.

i. **stp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping STP packets.

j. **Recovery Interval**—Enter the amount of time, in seconds, to wait before recovering a UNI port.

**Step 23** Click the **Next** button, if you want to enable template support for the policy.

The Template Association window appears. In this window, you can enable template support and, optionally, associate templates and data files with the policy. For instructions about associating templates with policies and how to use the features in this window, see Appendix B, "Working with Templates and Data Files". When you have completed setting up templates and data files for the policy, click **Finish** in the Template Association window to close it and return to the Policy Editor window.

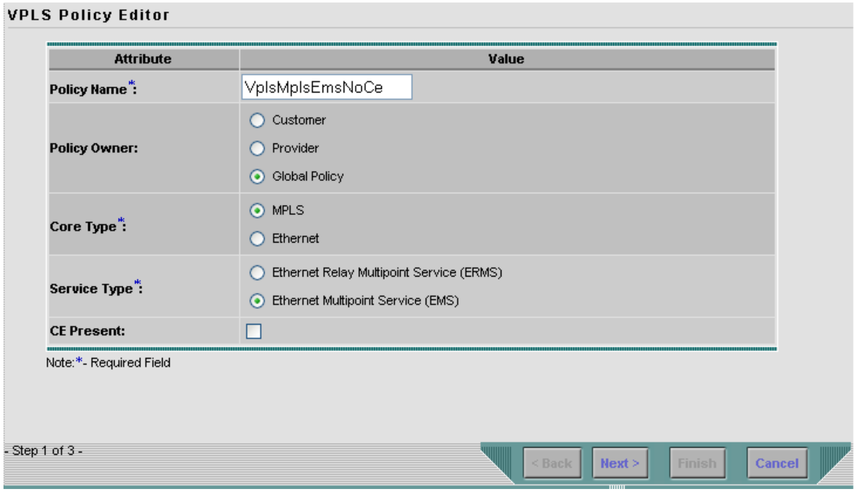
**Step 24** Click **Finish**.

**Note** The VC ID is mapped from the VPN ID. By default, ISC will "auto pick" this value. However, you can set this manually, if desired. This is done by editing the associated VPN configuration. The Edit VPN window has an **Enable VPLS** check box. When you check this box, you can manually enter a VPN ID in a field provided. For more information on creating and modifying VPNs, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*.

# Defining an MPLS/EMS (EP-LAN) Policy without a CE

This section describes defining a VPLS policy with an MPLS core type and an EMS (EP-LAN) service type without a CE present. Figure 9-15 is an example of the first page of this policy.

*Figure 9-15 MPLS/EMS (EP-LAN) Policy without a CE*



VPLS Policy Editor

| Attribute | Value |
|---|---|
| Policy Name*: | VplsMplsEmsNoCe |
| Policy Owner: | ○ Customer<br>○ Provider<br>● Global Policy |
| Core Type*: | ● MPLS<br>○ Ethernet |
| Service Type*: | ○ Ethernet Relay Multipoint Service (ERMS)<br>● Ethernet Multipoint Service (EMS) |
| CE Present: | □ |

Note:*- Required Field

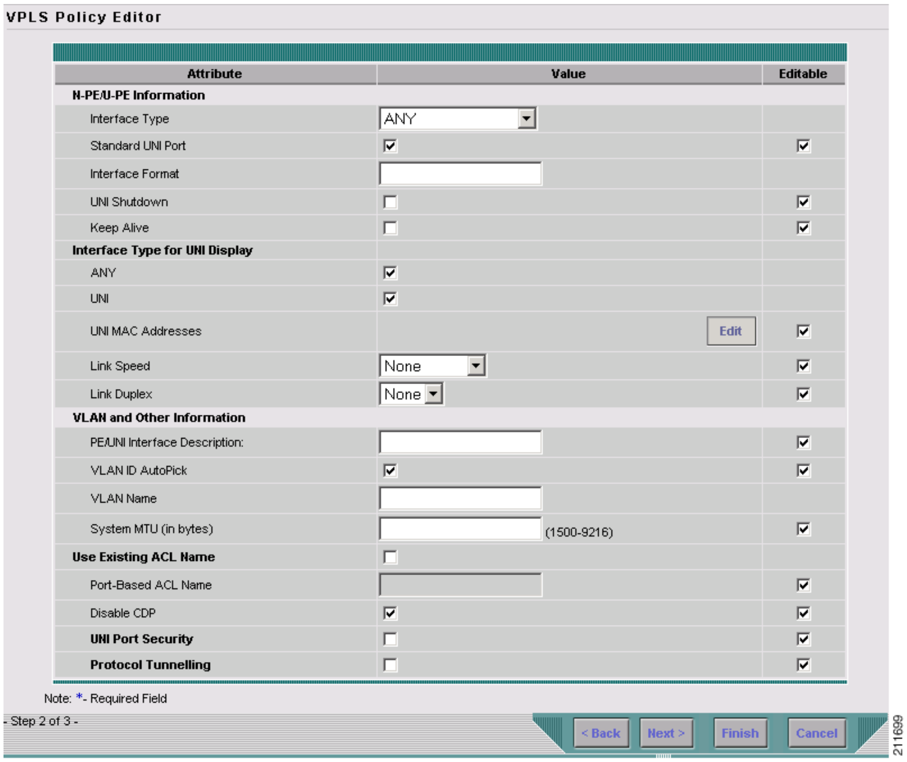
- Step 1 of 3 -

Perform the following steps.

**Step 1** Click **Next**. The window in Figure 9-16 appears.

The **Editable** check box gives you the option of making a field editable. If you check **Editable** check box, the service operator who is using this VPLS policy can modify the editable parameter during VPLS service request creation.

*Figure 9-16 MPLS/EMS (EP-LAN) without a CE Policy Attributes*



VPLS Policy Editor

| Attribute | Value | Editable |
|---|---|---|
| **N-PE/U-PE Information** | | |
| Interface Type | ANY ▾ | |
| Standard UNI Port | ☑ | ☑ |
| Interface Format | | |
| UNI Shutdown | □ | ☑ |
| Keep Alive | □ | ☑ |
| **Interface Type for UNI Display** | | |
| ANY | ☑ | |
| UNI | ☑ | |
| UNI MAC Addresses | Edit | ☑ |
| Link Speed | None ▾ | ☑ |
| Link Duplex | None ▾ | ☑ |
| **VLAN and Other Information** | | |
| PE/UNI Interface Description: | | ☑ |
| VLAN ID AutoPick | ☑ | ☑ |
| VLAN Name | | |
| System MTU (in bytes) | (1500-9216) | ☑ |
| **Use Existing ACL Name** | □ | |
| Port-Based ACL Name | | ☑ |
| Disable CDP | ☑ | ☑ |
| **UNI Port Security** | □ | ☑ |
| **Protocol Tunnelling** | □ | ☑ |

Note: *- Required Field

- Step 2 of 3 -

**Step 2** Choose an **Interface Type** from the drop-down list.

You can choose a particular interface on a N-PE, U-PE, or PE-AGG interface based on the service provider's POP design. The interfaces are:

- **ANY** (Any interface can be chosen.)
- **Port-Channel** (A bundle of ports that share the same characteristics—this gives the service provider the ability to aggregate bandwidth and protection.)
- **Ethernet**
- **FastEthernet**
- **GE-WAN**
- **GigabitEthernet**
- **TenGigabitEthernet**
- **TenGigE**

The value defined here functions as a filter to restrict the interface types an operator can see during VPLS service request creation. If defined as ANY, the operator can see all interface types.

**Step 3** Check the **Standard UNI Port** check box to enable port security.

This is the default. When you uncheck the check box, the port is treated as an uplink with no security features, and the window dynamically changes to eliminate items related to port security.

**Step 4** Enter an **Interface Format** as the slot number/port number for the PE interface (for example, **1/0** indicates that the interface is located at slot 1, port 0).

This is especially useful to specify here if you know that the link will always go through a particular interface's slot/port location on all or most of the network devices in the service.

**Step 5** Choose a N-PE/U-PE **Encapsulation** type.

The choices are:

- **DOT1Q**
- **DEFAULT**

**Note** When creating a service request based on the MPLS/EMS (EP-LAN) without CE policy, the Encapsulation attribute is ignored. Therefore, setting this value has no effect.

**Step 6** Check the **UNI Shutdown** check box if you want to leave the UNI port shut during service activation, for example, when the service provider wants to deploy a service in the network but wants to activate it at a later time.

**Step 7** Check the **Keep Alive** check box to configure keepalives on the UNI port.

By default, this check box is unchecked, which causes the command **no keepalive** to be provisioned on the UNI port. This prevents a CPE from sending keepalive packets to the U-PE, for security purposes. This attribute is editable to support modification on a per-service request basis.

**Step 8** Check the **ANY** check box to display all interface types as choices for the UNI interface (when creating service requests based on this policy).

This check box is checked by default.

**Step 9** Check the **UNI** check box to display all interfaces defined as type UNI as choices for the UNI interface (when creating service requests based on this policy). This check box is checked by default.

**Step 10** Enter one or more Ethernet MAC addresses in **UNI MAC addresses**.

This selection is present only if you uncheck the **Use Existing ACL Name** check box. Click the **Edit** button to bring up a pop-up window in which you enter MAC addresses to be allowed or denied on the port. You can also specify a range of addresses by setting a base MAC address and a filtered MAC address.

**Step 11** Enter a **Link Speed** (optional) of None, 10, 100, 1000, Auto, or nonegotiate.

**Step 12** Enter a **Link Duplex** (optional) of None, Full, Half, or Auto.

**Step 13** In the **PE/UNI Interface Description** field, enter an optional description, for example *Customer-B EMS (EP-LAN) Service*.

**Step 14** Check the **VLAN ID AutoPick** check box if you want ISC to choose a VLAN ID.

If you do not check this check box, you will be prompted to provide the VLAN in a Provider VLAN ID field during service activation.

**Step 15** Enter a **VLAN NAME** (optional) to specify a name to describe the VLAN.

The name must be one token (no spaces allowed.) The limit for the VLAN name is 32 characters. The name has to be unique. Two VLANs cannot share the same name.

**Step 16** Enter the **System MTU** in bytes.

The maximum transmission unit (MTU) size is configurable and optional. ISC does not perform an integrity check for this customized value. If a service request goes to the **Failed Deploy** state because this size is not accepted, you must adjust the size until the service request is deployed. ISC supports, ranges for different platforms, as specified below. The range is 1500 to 9216.

- For the 3750 and 3550 platforms, the MTU range is 1500-1546.
- For the 7600 ethernet port, the MTU size is always 9216. Even with the same platform and same IOS release, different line cards support the MTU differently. For example, older line cards only take an MTU size of 9216 and newer cards support 1500-9216. However, ISC uses 9216 in both cases.
- For the 7600 SVI (interface VLAN), the MTU size is 1500-9216.

**Step 17** Check the **Use Existing ACL Name** check box if you want assign your own named access list to the port.

By default, this check box is not checked and ISC automatically assigns a MAC-based ACL on the customer facing UNI port, based on values you enter in **UNI MAC addresses** (below).

**Step 18** Enter a **Port-Based ACL Name** (if you checked the **Use Existing ACL Name** check box, as mentioned in the previous step).

**Note** ISC does not create this ACL automatically. The ACL must already exist on the device, or be added as part of a template, before the service request is deployed. Otherwise, deployment will fail.

**Step 19** Check the **Disable CDP** check box if you want to disable the Cisco Discover Protocol (CDP) on the UNI port.

**Step 20** Check the **UNI Port Security** check box (see Figure 9-17) if you to want to provision port security-related CLIs to the UNI port by controlling the MAC addresses that are allowed to go through the interface.

a. For **Maximum Number of MAC address**, enter the number of MAC addresses allowed for port security.

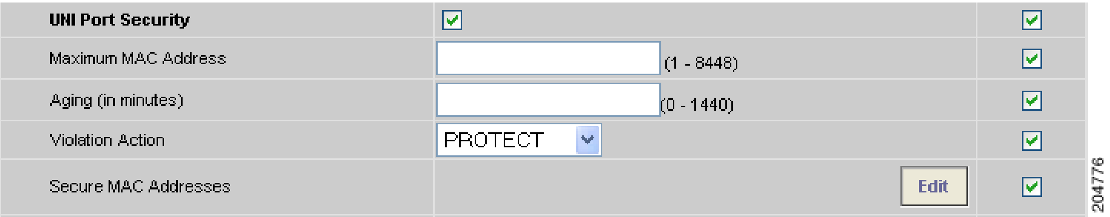
b. For **Aging,** enter the length of time the MAC address can stay on the port security table.

c. For **Violation Action**, choose what action will occur when a port security violation is detected:

- **PROTECT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value.
- **RESTRICT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value and causes the Security Violation counter to increment.
- **SHUTDOWN**—Puts the interface into the error-disabled state immediately and sends an SNMP trap notification.

d. In the **Secure MAC Addresses** field, enter one or more Ethernet MAC addresses.

*Figure 9-17 UNI Port Security*

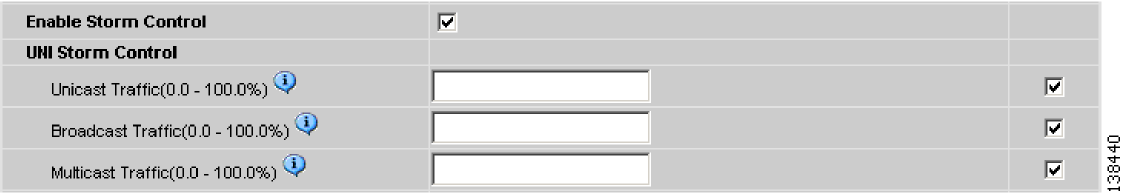


| | | |
|---|---|---|
| **UNI Port Security** | ☑ | ☑ |
| Maximum MAC Address | (1 - 8448) | ☑ |
| Aging (in minutes) | (0 - 1440) | ☑ |
| Violation Action | PROTECT | ☑ |
| Secure MAC Addresses | Edit | ☑ |

**Step 21** Check the **Enable Storm Control** check box (see Figure 9-18) to help prevent the UNI port from being disrupted by a broadcast, multicast or unicast storm.

Enter a threshold value for each type of traffic. The value, which can be specified to two significant digits, represents the percentage of the total available bandwidth of the port. If the threshold of a traffic type is reached, further traffic of that type is suppressed until the incoming traffic falls below the threshold level.

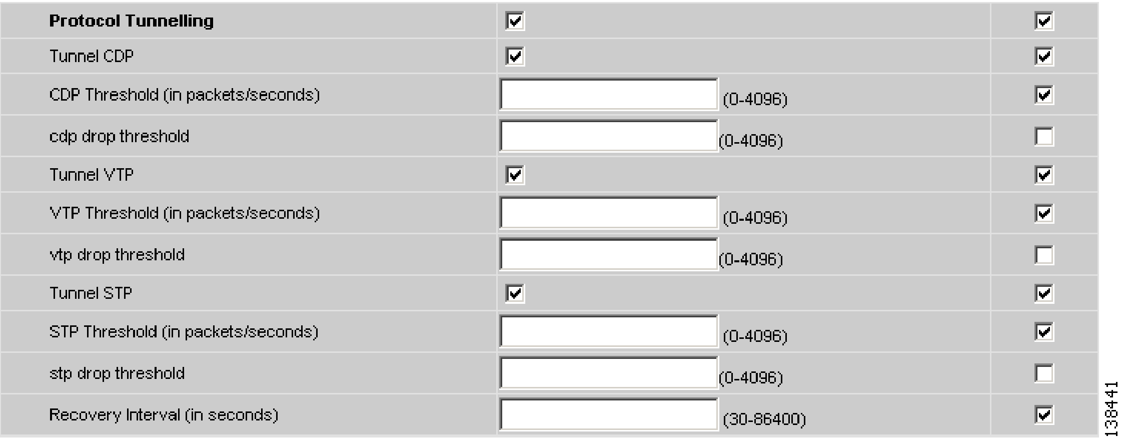
*Figure 9-18 Enable Storm Control*



| | | |
|---|---|---|
| **Enable Storm Control** | ☑ | |
| **UNI Storm Control** | | |
| Unicast Traffic(0.0 - 100.0%) | | ☑ |
| Broadcast Traffic(0.0 - 100.0%) | | ☑ |
| Multicast Traffic(0.0 - 100.0%) | | ☑ |

**Step 22** Check the **Protocol Tunnelling** check box (see Figure 9-19) if you want to define the Layer 2 Bridge Protocol Data Unit (BPDU) frames that can be tunneled over the core to the other end.

*Figure 9-19 Protocol Tunnelling*



| | | |
|---|---|---|
| **Protocol Tunnelling** | ☑ | ☑ |
| Tunnel CDP | ☑ | ☑ |
| CDP Threshold (in packets/seconds) | (0-4096) | ☑ |
| cdp drop threshold | (0-4096) | ☐ |
| Tunnel VTP | ☑ | ☑ |
| VTP Threshold (in packets/seconds) | (0-4096) | ☑ |
| vtp drop threshold | (0-4096) | ☐ |
| Tunnel STP | ☑ | ☑ |
| STP Threshold (in packets/seconds) | (0-4096) | ☑ |
| stp drop threshold | (0-4096) | ☐ |
| Recovery Interval (in seconds) | (30-86400) | ☑ |

For each protocol that you check, enter the shutdown threshold and drop threshold for that protocol:

a. **Tunnel CDP**—Enable Layer 2 tunnelling on Cisco Discover Protocol (CDP).

b. **CDP Threshold**—Enter the number of packets per second to be received before the interface is shut down.

c. **cdp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping CDP packets.

d. **Tunnel VTP**—Enable Layer 2 tunnelling on VLAN Trunk Protocol (VTP).

e. **VTP threshold**—Enter the number of packets per second to be received before the interface is shut down.

f. **vtp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping VTP packets.

g. **Tunnel STP**—Enable Layer 2 tunnelling on Spanning Tree Protocol (STP).

h. **STP Threshold**—Enter the number of packets per second to be received before the interface is shut down.

i. **stp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping STP packets.

j. **Recovery Interval**—Enter the amount of time, in seconds, to wait before recovering a UNI port.

**Step 23** Click the **Next** button, if you want to enable template support for the policy.

The Template Association window appears. In this window, you can enable template support and, optionally, associate templates and data files with the policy. For instructions about associating templates with policies and how to use the features in this window, see Appendix B, "Working with Templates and Data Files". When you have completed setting up templates and data files for the policy, click **Finish** in the Template Association window to close it and return to the Policy Editor window.

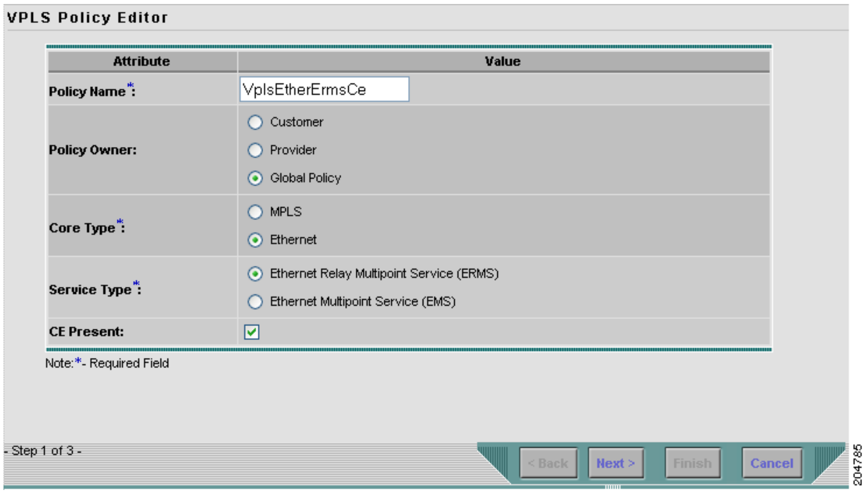
**Step 24** Click **Finish**.

**Note** The VC ID is mapped from the VPN ID. By default, ISC will "auto pick" this value. However, you can set this manually, if desired. This is done by editing the associated VPN configuration. The Edit VPN window has an **Enable VPLS** check box. When you check this box, you can manually enter a VPN ID in a field provided. For more information on creating and modifying VPNs, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*.

# Defining an Ethernet/ERMS (EVP-LAN) Policy with a CE

This section describes defining a VPLS policy with an Ethernet core type and an ERMS (EVP-LAN) service type with CE present. Figure 9-20 is an example of the first page of this policy.

*Figure 9-20 Ethernet/ERMS (EVP-LAN) Policy with a CE*



VPLS Policy Editor

| Attribute | Value |
|---|---|
| Policy Name*: | VplsEtherErmsCe |
| Policy Owner: | ○ Customer<br>○ Provider<br>● Global Policy |
| Core Type*: | ○ MPLS<br>● Ethernet |
| Service Type*: | ● Ethernet Relay Multipoint Service (ERMS)<br>○ Ethernet Multipoint Service (EMS) |
| CE Present: | ☑ |

Note:*- Required Field

- Step 1 of 3 -

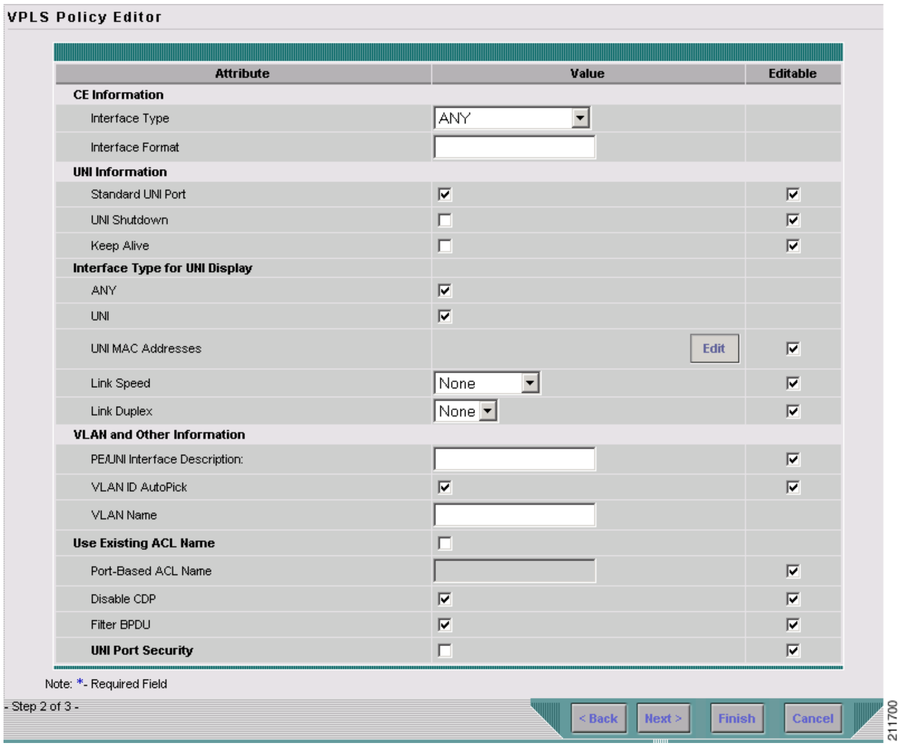
Perform the following steps.

**Step 1** Click **Next**.

The window in Figure 9-21 appears.

The **Editable** check box gives you the option of making a field editable. If you check the **Editable** check box, the service operator who is using this VPLS policy can modify the editable parameter during VPLS service request creation.

*Figure 9-21 Ethernet ERMS (EVP-LAN) with a CE Policy Attributes*



VPLS Policy Editor

| Attribute | Value | Editable |
|---|---|---|
| **CE Information** | | |
| Interface Type | ANY ▾ | |
| Interface Format | | |
| **UNI Information** | | |
| Standard UNI Port | ☑ | ☑ |
| UNI Shutdown | □ | ☑ |
| Keep Alive | □ | ☑ |
| **Interface Type for UNI Display** | | |
| ANY | ☑ | |
| UNI | ☑ | |
| UNI MAC Addresses | Edit | ☑ |
| Link Speed | None ▾ | ☑ |
| Link Duplex | None ▾ | ☑ |
| **VLAN and Other Information** | | |
| PE/UNI Interface Description: | | ☑ |
| VLAN ID AutoPick | ☑ | ☑ |
| VLAN Name | | |
| **Use Existing ACL Name** | □ | |
| Port-Based ACL Name | | ☑ |
| Disable CDP | ☑ | ☑ |
| Filter BPDU | ☑ | ☑ |
| **UNI Port Security** | □ | ☑ |

Note: *- Required Field

- Step 2 of 3 -

**Step 2** Choose an **Interface Type** from the drop-down list.

You can choose a particular interface on a CE, N-PE, U-PE, or PE-AGG interface based on the service provider’s POP design. The interfaces are:

- **ANY** (Any interface can be chosen.)
- **Port-Channel** (A bundle of ports that share the same characteristics—this gives the service provider the ability to aggregate bandwidth and protection.)
- **Ethernet**
- **FastEthernet**
- **GE-WAN**
- **GigabitEthernet**
- **TenGigabitEthernet**
- **TenGigE**

The value defined here functions as a filter to restrict the interface types an operator can see during VPLS service request creation. If defined as ANY, the operator can see all interface types.

**Step 3** Enter an **Interface Format** as the slot number/port number for the CE interface (for example, **1/0** indicates that the interface is located at slot 1, port 0).

This is especially useful to specify here if you know that the link will always go through a particular interface’s slot/port location on all or most of the network devices in the service.

**Step 4** Choose a CE **Encapsulation** type.

The choices are:

- **DOT1Q**
- **DEFAULT**

If **DEFAULT** is the CE encapsulation type, ISC shows another field for the UNI port type.

**Step 5** Check the **Standard UNI Port** check box to enable port security.

This is the default. When you uncheck the check box, the port is treated as an uplink with no security features, and the window dynamically changes to eliminate items related to port security.

**Step 6** Check the **UNI Shutdown** check box if you want to leave the UNI port shut during service activation, for example, when the service provider wants to deploy a service in the network but wants to activate it at a later time.

**Step 7** Check the **Keep Alive** check box to configure keepalives on the UNI port.

By default, this check box is unchecked, which causes the command **no keepalive** to be provisioned on the UNI port. This prevents a CPE from sending keepalive packets to the U-PE, for security purposes. This attribute is editable to support modification on a per-service request basis.

**Step 8** Check the **ANY** check box to display all interface types as choices for the UNI interface (when creating service requests based on this policy).

This check box is checked by default.

**Step 9** Check the **UNI** check box to display all interfaces defined as type UNI as choices for the UNI interface (when creating service requests based on this policy).

This check box is checked by default.

**Step 10** Enter one or more Ethernet MAC addresses in **UNI MAC addresses**.

This selection is present only if you uncheck the **Use Existing ACL Name** check box. Click the **Edit** button to bring up a pop-up window in which you enter MAC addresses to be allowed or denied on the port. You can also specify a range of addresses by setting a base MAC address and a filtered MAC address.

**Step 11** Choose a **Port Type**.

The choices are:

- **Access Port**
- **Trunk with Native VLAN**

**Step 12** Enter a **Link Speed** (optional) of None, 10, 100, 1000, Auto, or nonegotiate.

**Step 13** Enter a **Link Duplex** (optional) of None, Full, Half, or Auto.

**Step 14** In the **PE/UNI Interface Description** field, enter an optional description, for example *Customer-B ERMS (EVP-LAN) Service*.

**Step 15** Check the **VLAN ID AutoPick** check box if you want ISC to choose a VLAN ID.

If you do not check this check box, you will be prompted to provide the VLAN in a Provider VLAN ID field during service activation.

**Step 16** Enter a **VLAN NAME** (optional) to specify a name to describe the VLAN.

The name must be one token (no spaces allowed.) The limit for the VLAN name is 32 characters. The name has to be unique. Two VLANs cannot share the same name.

**Step 17** Check the **Use Existing ACL Name** check box if you want assign your own named access list to the port.

By default, this check box is not checked and ISC automatically assigns a MAC-based ACL on the customer facing UNI port, based on values you enter in **UNI MAC addresses** (below).

**Step 18** Enter a **Port-Based ACL Name** (if you checked the **Use Existing ACL Name** check box, as mentioned in the previous step).

**Note** ISC does not create this ACL automatically. The ACL must already exist on the device, or be added as part of a template, before the service request is deployed. Otherwise, deployment will fail.

**Step 19** Check the **Disable CDP** check box if you want to disable the Cisco Discover Protocol (CDP) on the UNI port.

**Step 20** Check the **Filter BPDU** check box to specify that the UNI port should not process Layer 2 Bridge Protocol Data Units (BPDUs).

**Step 21** Check the **UNI Port Security** check box (see Figure 9-22) if you to want to provision port security-related CLIs to the UNI port by controlling the MAC addresses that are allowed to go through the interface.

**a.** For **Maximum Number of MAC address**, enter the number of MAC addresses allowed for port security.

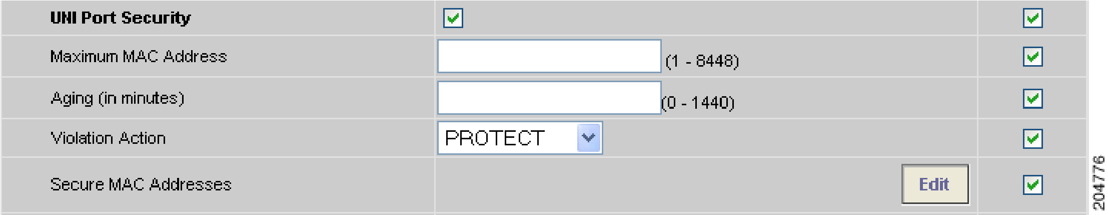
**b.** For **Aging,** enter the length of time the MAC address can stay on the port security table.

**c.** For **Violation Action**, choose what action will occur when a port security violation is detected:

- **PROTECT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value.
- **RESTRICT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value and causes the Security Violation counter to increment.
- **SHUTDOWN**—Puts the interface into the error-disabled state immediately and sends an SNMP trap notification.

**d.** In the **Secure MAC Addresses** field, enter one or more Ethernet MAC addresses.

***Figure 9-22 UNI Port Security***

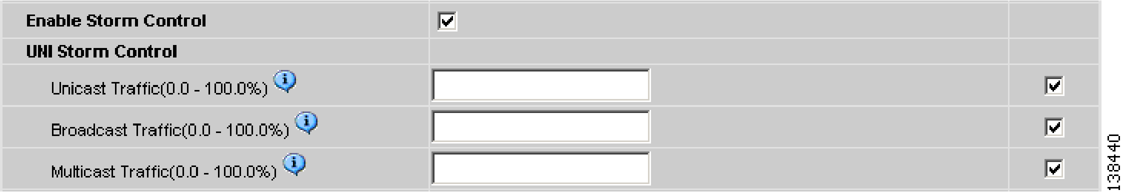


| | | |
|---|---|---|
| **UNI Port Security** | ☑ | ☑ |
| Maximum MAC Address | (1 - 8448) | ☑ |
| Aging (in minutes) | (0 - 1440) | ☑ |
| Violation Action | PROTECT | ☑ |
| Secure MAC Addresses | Edit | ☑ |

**Step 22** Check the **Enable Storm Control** check box (see Figure 9-22) to help prevent the UNI port from being disrupted by a broadcast, multicast or unicast storm.

Enter a threshold value for each type of traffic. The value, which can be specified to two significant digits, represents the percentage of the total available bandwidth of the port. If the threshold of a traffic type is reached, further traffic of that type is suppressed until the incoming traffic falls below the threshold level.

*Figure 9-23 Enable Storm Control*



| | | |
|---|---|---|
| **Enable Storm Control** | ☑ | |
| **UNI Storm Control** | | |
| Unicast Traffic(0.0 - 100.0%) ⓘ | | ☑ |
| Broadcast Traffic(0.0 - 100.0%) ⓘ | | ☑ |
| Multicast Traffic(0.0 - 100.0%) ⓘ | | ☑ |

**Step 23** Click the **Next** button, if you want to enable template support for the policy.

The Template Association window appears. In this window, you can enable template support and, optionally, associate templates and data files with the policy. For instructions about associating templates with policies and how to use the features in this window, see Appendix B, "Working with Templates and Data Files". When you have completed setting up templates and data files for the policy, click **Finish** in the Template Association window to close it and return to the Policy Editor window.

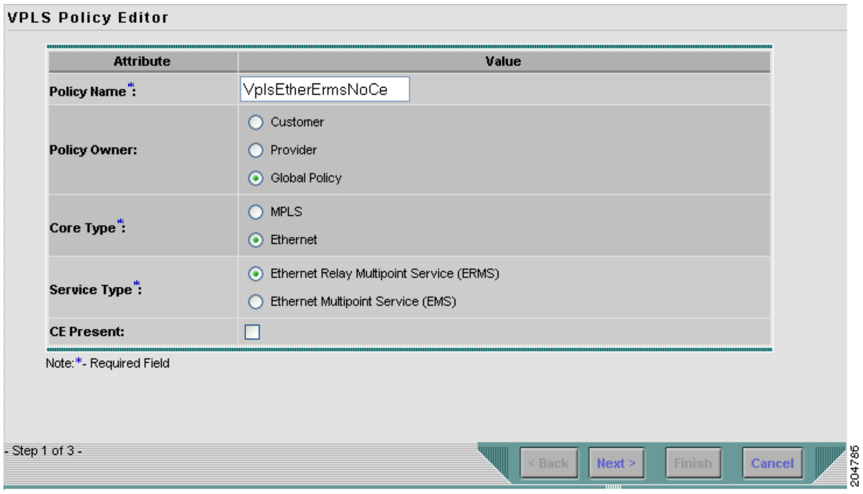
**Step 24** Click **Finish**.

**Note** The VC ID is mapped from the VPN ID. By default, ISC will "auto pick" this value. However, you can set this manually, if desired. This is done by editing the associated VPN configuration. The Edit VPN window has an **Enable VPLS** check box. When you check this box, you can manually enter a VPN ID in a field provided. For more information on creating and modifying VPNs, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*.

# Defining an Ethernet/ERMS (EVP-LAN) Policy without a CE

This section describes defining a VPLS policy with an Ethernet core type and an ERMS (EVP-LAN) service type without a CE present. Figure 9-24 is an example of the first page of this policy.

*Figure 9-24 Ethernet/ERMS (EVP-LAN) Policy without a CE*



VPLS Policy Editor

| Attribute | Value |
|---|---|
| **Policy Name*:** | VplsEtherErmsNoCe |
| **Policy Owner:** | ○ Customer<br>○ Provider<br>● Global Policy |
| **Core Type*:** | ○ MPLS<br>● Ethernet |
| **Service Type*:** | ● Ethernet Relay Multipoint Service (ERMS)<br>○ Ethernet Multipoint Service (EMS) |
| **CE Present:** | □ |

Note:*- Required Field

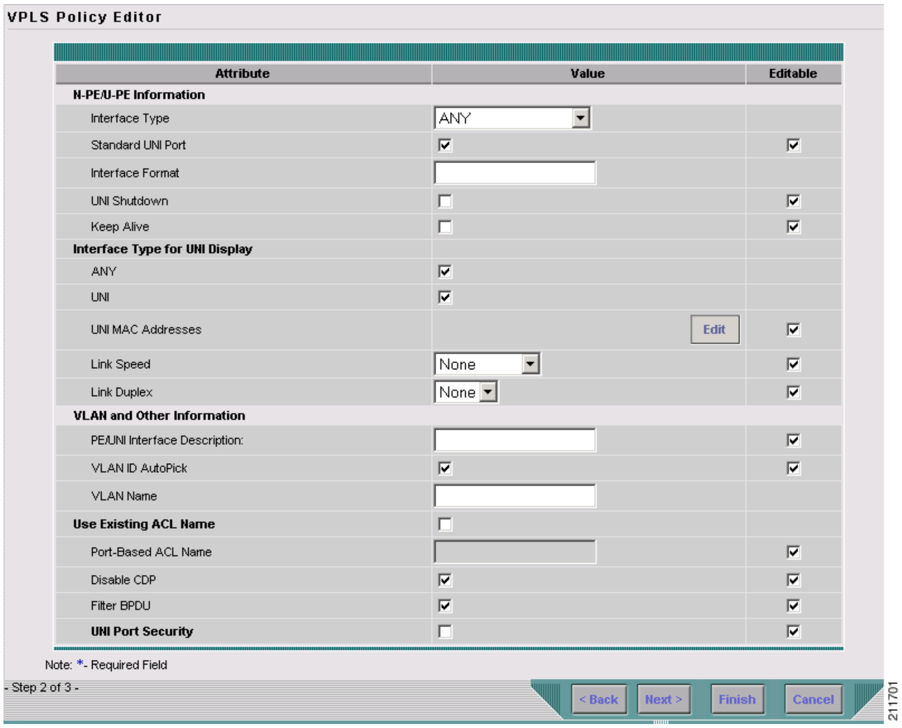
Perform the following steps.

**Step 1** Click **Next**.

The window in Figure 9-25 appears.

The **Editable** check box gives you the option of making a field editable. If you check the **Editable** check box, the service operator who is using this VPLS policy can modify the editable parameter during VPLS service request creation.

***Figure 9-25 Ethernet/ERMS (EVP-LAN) without a CE Policy Attributes***



VPLS Policy Editor

| Attribute | Value | Editable |
|---|---|---|
| **N-PE/U-PE Information** | | |
| Interface Type | ANY ▾ | |
| Standard UNI Port | ☑ | ☑ |
| Interface Format | | |
| UNI Shutdown | □ | ☑ |
| Keep Alive | □ | ☑ |
| **Interface Type for UNI Display** | | |
| ANY | ☑ | |
| UNI | ☑ | |
| UNI MAC Addresses | Edit | ☑ |
| Link Speed | None ▾ | ☑ |
| Link Duplex | None ▾ | ☑ |
| **VLAN and Other Information** | | |
| PE/UNI Interface Description: | | ☑ |
| VLAN ID AutoPick | ☑ | ☑ |
| VLAN Name | | |
| **Use Existing ACL Name** | □ | |
| Port-Based ACL Name | | ☑ |
| Disable CDP | ☑ | ☑ |
| Filter BPDU | ☑ | ☑ |
| **UNI Port Security** | □ | ☑ |

Note: *- Required Field

- Step 2 of 3 -

**Step 2** Choose an **Interface Type** from the drop-down list.

You can choose a particular interface on a CE, N-PE, U-PE, or PE-AGG interface based on the service provider's POP design. The interfaces are:

- **ANY** (Any interface can be chosen.)
- **Port-Channel** (A bundle of ports that share the same characteristics—this gives the service provider the ability to aggregate bandwidth and protection.)
- **Ethernet**
- **FastEthernet**
- **GE-WAN**
- **GigabitEthernet**
- **TenGigabitEthernet**
- **TenGigE**

The value defined here functions as a filter to restrict the interface types an operator can see during VPLS service request creation. If defined as ANY, the operator can see all interface types.

**Step 3** Check the **Standard UNI Port** check box to enable port security.

This is the default. When you uncheck the check box, the port is treated as an uplink with no security features, and the window dynamically changes to eliminate items related to port security.

**Step 4** Enter an **Interface Format** as the slot number/port number for the CE interface (for example, **1/0** indicates that the interface is located at slot 1, port 0).

This is especially useful to specify here if you know that the link will always go through a particular interface's slot/port location on all or most of the network devices in the service.

**Step 5** Choose a CE **Encapsulation** type.

The choices are:

- **DOT1Q**
- **DEFAULT**

If **DEFAULT** is the CE encapsulation type, ISC shows another field for the UNI port type.

**Step 6** Check the **UNI Shutdown** check box if you want to leave the UNI port shut during service activation, for example, when the service provider wants to deploy a service in the network but wants to activate it at a later time.

**Step 7** Check the **Keep Alive** check box to configure keepalives on the UNI port.

By default, this check box is unchecked, which causes the command **no keepalive** to be provisioned on the UNI port. This prevents a CPE from sending keepalive packets to the U-PE, for security purposes. This attribute is editable to support modification on a per-service request basis.

**Step 8** Check the **ANY** check box to display all interface types as choices for the UNI interface (when creating service requests based on this policy).

This check box is checked by default.

**Step 9** Check the **UNI** check box to display all interfaces defined as type UNI as choices for the UNI interface (when creating service requests based on this policy). This check box is checked by default.

**Step 10** Enter one or more Ethernet MAC addresses in **UNI MAC addresses**.

This selection is present only if you uncheck the **Use Existing ACL Name** check box. Click the **Edit** button to bring up a pop-up window in which you enter MAC addresses to be allowed or denied on the port. You can also specify a range of addresses by setting a base MAC address and a filtered MAC address.

**Step 11** Choose a **Port Type**.

The choices are:

- **Access Port**
- **Trunk with Native VLAN**

**Step 12** Enter a **Link Speed** (optional) of None, 10, 100, 1000, Auto, or nonegotiate.

**Step 13** Enter a **Link Duplex** (optional) of None, Full, Half, or Auto.

**Step 14** In the **PE/UNI Interface Description** field, enter an optional description, for example *Customer-B ERMS (EVP-LAN) Service*.

**Step 15** Check the **VLAN ID AutoPick** check box if you want ISC to choose a VLAN ID.

If you do not check this check box, you will be prompted to provide the VLAN in a Provider VLAN ID field during service activation.

**Step 16** Enter a **VLAN NAME** (optional) to specify a name to describe the VLAN.

The name must be one token (no spaces allowed.) The limit for the VLAN name is 32 characters. The name has to be unique. Two VLANs cannot share the same name.

**Step 17** Check the **Use Existing ACL Name** check box if you want assign your own named access list to the port.

By default, this check box is not checked and ISC automatically assigns a MAC-based ACL on the customer facing UNI port, based on values you enter in **UNI MAC addresses** (below).

**Step 18** Enter a **Port-Based ACL Name** (if you checked the **Use Existing ACL Name** check box, as mentioned in the previous step).

**Note** ISC does not create this ACL automatically. The ACL must already exist on the device, or be added as part of a template, before the service request is deployed. Otherwise, deployment will fail.

**Step 19** Check the **Disable CDP** check box if you want to disable the Cisco Discover Protocol (CDP) on the UNI port.

**Step 20** Check the **Filter BPDU** check box to specify that the UNI port should not process Layer 2 Bridge Protocol Data Units (BPDUs).

**Step 21** Check the **UNI Port Security** check box (see Figure 9-26) if you to want to provision port security-related CLIs to the UNI port by controlling the MAC addresses that are allowed to go through the interface.

a. For **Maximum Number of MAC address**, enter the number of MAC addresses allowed for port security.

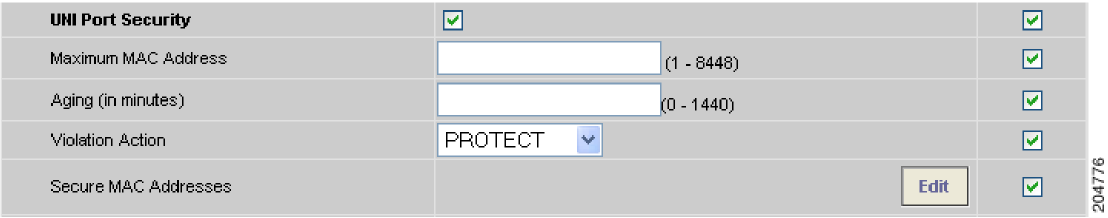
b. For **Aging,** enter the length of time the MAC address can stay on the port security table.

c. For **Violation Action**, choose what action will occur when a port security violation is detected:

- **PROTECT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value.
- **RESTRICT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value and causes the Security Violation counter to increment.
- **SHUTDOWN**—Puts the interface into the error-disabled state immediately and sends an SNMP trap notification.

d. In the **Secure MAC Addresses** field, enter one or more Ethernet MAC addresses.

*Figure 9-26 UNI Port Security*

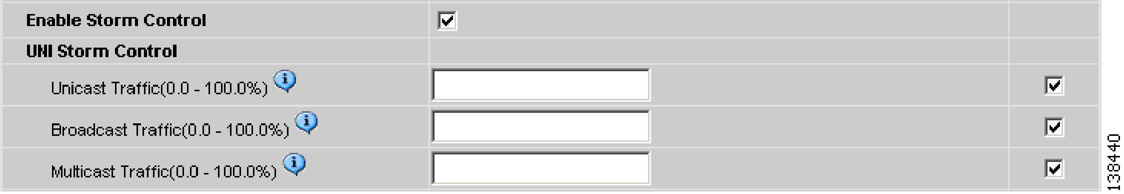


| | | |
|---|---|---|
| **UNI Port Security** | ☑ | ☑ |
| Maximum MAC Address | (1 - 8448) | ☑ |
| Aging (in minutes) | (0 - 1440) | ☑ |
| Violation Action | PROTECT | ☑ |
| Secure MAC Addresses | Edit | ☑ |

**Step 22** Check the **Enable Storm Control** check box (see Figure 9-27) to help prevent the UNI port from being disrupted by a broadcast, multicast or unicast storm.

Enter a threshold value for each type of traffic. The value, which can be specified to two significant digits, represents the percentage of the total available bandwidth of the port. If the threshold of a traffic type is reached, further traffic of that type is suppressed until the incoming traffic falls below the threshold level.

*Figure 9-27 Enable Storm Control*



| | | |
|---|---|---|
| **Enable Storm Control** | ☑ | |
| **UNI Storm Control** | | |
| Unicast Traffic(0.0 - 100.0%) ⓘ | | ☑ |
| Broadcast Traffic(0.0 - 100.0%) ⓘ | | ☑ |
| Multicast Traffic(0.0 - 100.0%) ⓘ | | ☑ |

**Step 23** Click the **Next** button, if you want to enable template support for the policy.

The Template Association window appears. In this window, you can enable template support and, optionally, associate templates and data files with the policy. For instructions about associating templates with policies and how to use the features in this window, see Appendix B, "Working with Templates and Data Files". When you have completed setting up templates and data files for the policy, click **Finish** in the Template Association window to close it and return to the Policy Editor window.

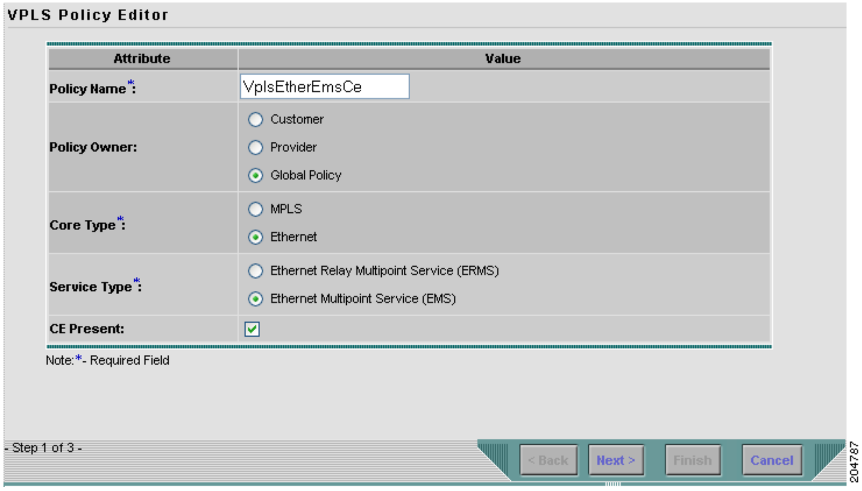
**Step 24** Click **Finish**.

**Note** The VC ID is mapped from the VPN ID. By default, ISC will "auto pick" this value. However, you can set this manually, if desired. This is done by editing the associated VPN configuration. The Edit VPN window has an **Enable VPLS** check box. When you check this box, you can manually enter a VPN ID in a field provided. For more information on creating and modifying VPNs, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*.

# Defining an Ethernet/EMS (EP-LAN) Policy with a CE

This section describes defining a VPLS policy with an Ethernet core type and an EMS (EP-LAN) service type with a CE present. Figure 9-28 is an example of the first page of this policy.

***Figure 9-28 Ethernet/EMS (EP-LAN) Policy with CE Present***



VPLS Policy Editor

| Attribute | Value |
|---|---|
| Policy Name*: | VplsEtherEmsCe |
| Policy Owner: | ○ Customer<br>○ Provider<br>● Global Policy |
| Core Type*: | ○ MPLS<br>● Ethernet |
| Service Type*: | ○ Ethernet Relay Multipoint Service (ERMS)<br>● Ethernet Multipoint Service (EMS) |
| CE Present: | ☑ |

Note:*- Required Field

- Step 1 of 3 -

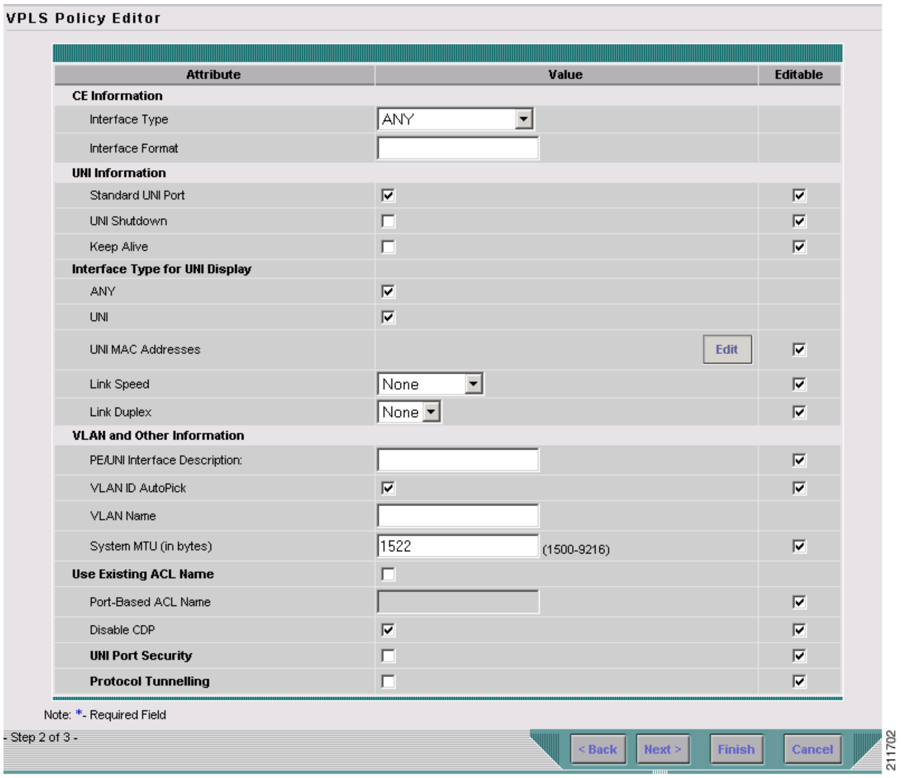
Perform the following steps.

**Step 1** Click **Next**.

The window in Figure 9-29 appears.

The **Editable** check box gives you the option of making a field editable. If you check the **Editable** check box, the service operator who is using this VPLS policy can modify the editable parameter during VPLS service request creation.

*Figure 9-29 Ethernet/EMS (EP-LAN) with a CE Policy Attributes*



VPLS Policy Editor

| Attribute | Value | Editable |
|---|---|---|
| **CE Information** | | |
| Interface Type | ANY ▾ | |
| Interface Format | | |
| **UNI Information** | | |
| Standard UNI Port | ☑ | ☑ |
| UNI Shutdown | □ | ☑ |
| Keep Alive | □ | ☑ |
| **Interface Type for UNI Display** | | |
| ANY | ☑ | |
| UNI | ☑ | |
| UNI MAC Addresses | Edit | ☑ |
| Link Speed | None ▾ | ☑ |
| Link Duplex | None ▾ | ☑ |
| **VLAN and Other Information** | | |
| PE/UNI Interface Description: | | ☑ |
| VLAN ID AutoPick | ☑ | ☑ |
| VLAN Name | | |
| System MTU (in bytes) | 1522 (1500-9216) | ☑ |
| **Use Existing ACL Name** | □ | |
| Port-Based ACL Name | | ☑ |
| Disable CDP | ☑ | ☑ |
| **UNI Port Security** | □ | ☑ |
| **Protocol Tunnelling** | □ | ☑ |

Note: *- Required Field

- Step 2 of 3 -

**Step 2** Choose an **Interface Type** from the drop-down list.

You can choose a particular interface on a CE, N-PE, U-PE, or PE-AGG interface based on the service provider's POP design.

The interfaces are:

- **ANY** (Any interface can be chosen.)
- **Port-Channel** (A bundle of ports that share the same characteristics—this gives the service provider the ability to aggregate bandwidth and protection.)
- **Ethernet**
- **FastEthernet**
- **GE-WAN**
- **GigabitEthernet**
- **TenGigabitEthernet**

The value defined here functions as a filter to restrict the interface types an operator can see during VPLS service request creation. If defined as ANY, the operator can see all interface types.

**Step 3** Enter an **Interface Format** as the slot number/port number for the CE interface (for example, **1/0** indicates that the interface is located at slot 1, port 0).

This is especially useful to specify here if you know that the link will always go through a particular interface's slot/port location on all or most of the network devices in the service.

**Step 4** Choose a CE **Encapsulation** type.

The choices are:

- **DOT1Q**
- **DEFAULT**

**Note** When creating a service request based on the Ethernet/EMS (EP-LAN) with CE policy, the Encapsulation attribute is ignored. Therefore, setting this value has no effect.

**Step 5** Check the **Standard UNI Port** check box to enable port security.

This is the default. When you uncheck the check box, the port is treated as an uplink with no security features, and the window dynamically changes to eliminate items related to port security.

**Step 6** Check the **UNI Shutdown** check box if you want to leave the UNI port shut during service activation, for example, when the service provider wants to deploy a service in the network but wants to activate it at a later time.

**Step 7** Check the **Keep Alive** check box to configure keepalives on the UNI port.

By default, this check box is unchecked, which causes the command **no keepalive** to be provisioned on the UNI port. This prevents a CPE from sending keepalive packets to the U-PE, for security purposes. This attribute is editable to support modification on a per-service request basis.

**Step 8** Check the **ANY** check box to display all interface types as choices for the UNI interface (when creating service requests based on this policy).

This check box is checked by default.

**Step 9** Check the **UNI** check box to display all interfaces defined as type UNI as choices for the UNI interface (when creating service requests based on this policy).

This check box is checked by default.

**Step 10** Enter one or more Ethernet MAC addresses in **UNI MAC addresses**.

This selection is present only if you uncheck the **Use Existing ACL Name** check box. Click the **Edit** button to bring up a pop-up window in which you enter MAC addresses to be allowed or denied on the port. You can also specify a range of addresses by setting a base MAC address and a filtered MAC address.

**Step 11** Enter a **Link Speed** (optional) of None, 10, 100, 1000, Auto, or nonegotiate.

**Step 12** Enter a **Link Duplex** (optional) of None, Full, Half, or Auto.

**Step 13** In the **PE/UNI Interface Description** field, enter an optional description, for example *Customer-B EMS (EP-LAN) Service*.

**Step 14** Check the **VLAN ID AutoPick** check box if you want ISC to choose a VLAN ID.

If you do not check this check box, you will be prompted to provide the VLAN in a Provider VLAN ID field during service activation.

**Step 15** Enter a **VLAN NAME** (optional) to specify a name to describe the VLAN.

The name must be one token (no spaces allowed.) The limit for the VLAN name is 32 characters. The name has to be unique. Two VLANs cannot share the same name.

**Step 16** Enter the **System MTU** in bytes.

The maximum transmission unit (MTU) size is configurable and optional. The default size is 9216, and the range is 1500 to 9216. ISC does not perform an integrity check for this customized value. If a service request goes to the Failed Deploy state because this size is not accepted, you must adjust the size until the Service Request is deployed.

In ISC 6.0, different platforms support different ranges.

- For the 3750 and 3550 platforms, the MTU range is 1500-1546.
- For the 7600 ethernet port, the MTU size is always 9216. Even with the same platform and same IOS release, different line cards support the MTU differently. For example, older line cards only take an MTU size of 9216 and newer cards support 1500-9216. However, ISC 6.0 uses 9216 in both cases.
- For the 7600 SVI (interface VLAN), the MTU size is 1500-9216.

**Step 17** Check the **Use Existing ACL Name** check box if you want assign your own named access list to the port.

By default, this is not checked and ISC automatically assigns a MAC-based ACL on the customer facing UNI port, based on values you enter in **UNI MAC addresses** (below).

**Step 18** Enter a **Port-Based ACL Name** (if you checked the **Use Existing ACL Name** check box, as mentioned in the previous step).

**Note** ISC does not create this ACL automatically. The ACL must already exist on the device, or be added as part of a template, before the service request is deployed. Otherwise, deployment will fail.

**Step 19** Check the **Disable CDP** check box if you want to disable the Cisco Discover Protocol (CDP) on the UNI port.

**Step 20** Check the **UNI Port Security** check box (see Figure 9-30) if you to want to provision port security-related CLIs to the UNI port by controlling the MAC addresses that are allowed to go through the interface.

a. For **Maximum Number of MAC address**, enter the number of MAC addresses allowed for port security.

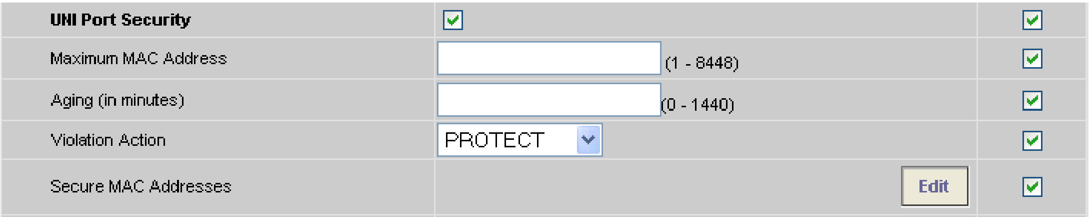
b. For **Aging,** enter the length of time the MAC address can stay on the port security table.

c. For **Violation Action**, choose what action will occur when a port security violation is detected:

- **PROTECT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value.
- **RESTRICT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value and causes the Security Violation counter to increment.
- **SHUTDOWN**—Puts the interface into the error-disabled state immediately and sends an SNMP trap notification.

d. In the **Secure MAC Addresses** field, enter one or more Ethernet MAC addresses.

*Figure 9-30 UNI Port Security*

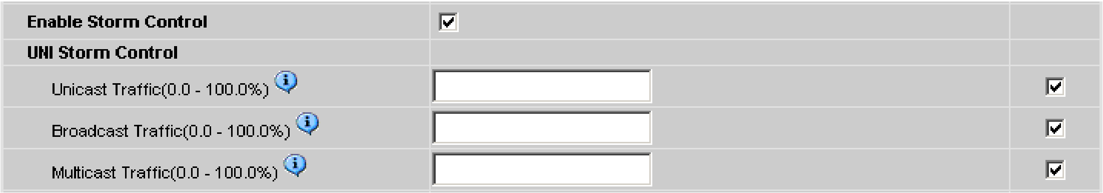


| | | |
|---|---|---|
| **UNI Port Security** | ☑ | ☑ |
| Maximum MAC Address | (1 - 8448) | ☑ |
| Aging (in minutes) | (0 - 1440) | ☑ |
| Violation Action | PROTECT | ☑ |
| Secure MAC Addresses | Edit | ☑ |

**Step 21** Check the **Enable Storm Control** check box (see Figure 9-31) to help prevent the UNI port from being disrupted by a broadcast, multicast or unicast storm.

Enter a threshold value for each type of traffic. The value, which can be specified to two significant digits, represents the percentage of the total available bandwidth of the port. If the threshold of a traffic type is reached, further traffic of that type is suppressed until the incoming traffic falls below the threshold level.

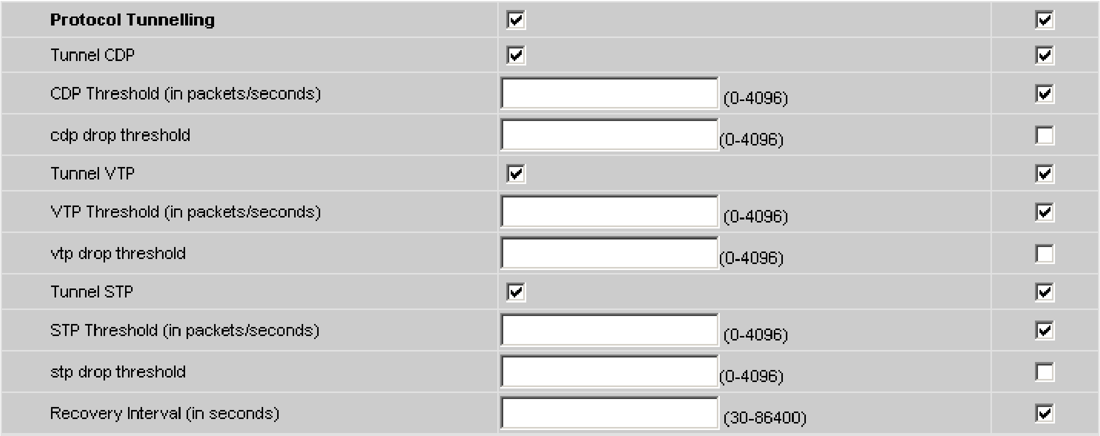
*Figure 9-31 Enable Storm Control*



| | | |
|---|---|---|
| **Enable Storm Control** | ☑ | |
| **UNI Storm Control** | | |
| Unicast Traffic(0.0 - 100.0%) | | ☑ |
| Broadcast Traffic(0.0 - 100.0%) | | ☑ |
| Multicast Traffic(0.0 - 100.0%) | | ☑ |

**Step 22** Check the **Protocol Tunnelling** check box (see Figure 9-32) if you want to define the Layer 2 Bridge Protocol Data Unit (BPDU) frames that can be tunneled over the core to the other end.

*Figure 9-32 Protocol Tunnelling*



| | | |
|---|---|---|
| **Protocol Tunnelling** | ☑ | ☑ |
| Tunnel CDP | ☑ | ☑ |
| CDP Threshold (in packets/seconds) | (0-4096) | ☑ |
| cdp drop threshold | (0-4096) | ☐ |
| Tunnel VTP | ☑ | ☑ |
| VTP Threshold (in packets/seconds) | (0-4096) | ☑ |
| vtp drop threshold | (0-4096) | ☐ |
| Tunnel STP | ☑ | ☑ |
| STP Threshold (in packets/seconds) | (0-4096) | ☑ |
| stp drop threshold | (0-4096) | ☐ |
| Recovery Interval (in seconds) | (30-86400) | ☑ |

For each protocol that you check, enter the shutdown threshold and drop threshold for that protocol:

**a.** **Tunnel CDP**—Enable Layer 2 tunnelling on Cisco Discover Protocol (CDP).

**b.** **CDP Threshold**—Enter the number of packets per second to be received before the interface is shut down.

c. **cdp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping CDP packets.

d. **Tunnel VTP**—Enable Layer 2 tunnelling on VLAN Trunk Protocol (VTP).

e. **VTP threshold**—Enter the number of packets per second to be received before the interface is shut down.

f. **vtp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping VTP packets.

g. **Tunnel STP**—Enable Layer 2 tunnelling on Spanning Tree Protocol (STP).

h. **STP Threshold**—Enter the number of packets per second to be received before the interface is shut down.

i. **stp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping STP packets.

j. **Recovery Interval**—Enter the amount of time, in seconds, to wait before recovering a UNI port.

**Step 23** Click the **Next** button, if you want to enable template support for the policy.

The Template Association window appears. In this window, you can enable template support and, optionally, associate templates and data files with the policy. For instructions about associating templates with policies and how to use the features in this window, see Appendix B, "Working with Templates and Data Files". When you have completed setting up templates and data files for the policy, click **Finish** in the Template Association window to close it and return to the Policy Editor window.

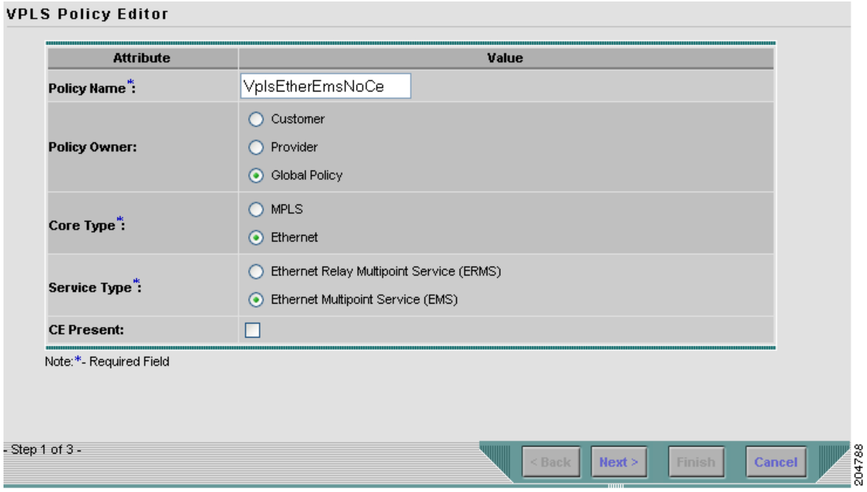
**Step 24** Click **Finish**.

**Note** The VC ID is mapped from the VPN ID. By default, ISC will "auto pick" this value. However, you can set this manually, if desired. This is done by editing the associated VPN configuration. The Edit VPN window has an **Enable VPLS** check box. When you check this box, you can manually enter a VPN ID in a field provided. For more information on creating and modifying VPNs, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*.

# Defining an Ethernet/EMS (EP-LAN) Policy without a CE

This section describes defining a VPLS policy with an Ethernet core type and an EMS (EP-LAN) service type without a CE present. Figure 9-33 is an example of the first page of this policy.

*Figure 9-33 Ethernet/EMS (EP-LAN) Policy without a CE*



VPLS Policy Editor

| Attribute | Value |
|---|---|
| Policy Name*: | VplsEtherEmsNoCe |
| Policy Owner: | ○ Customer<br>○ Provider<br>● Global Policy |
| Core Type*: | ○ MPLS<br>● Ethernet |
| Service Type*: | ○ Ethernet Relay Multipoint Service (ERMS)<br>● Ethernet Multipoint Service (EMS) |
| CE Present: | □ |

Note:*- Required Field

- Step 1 of 3 -

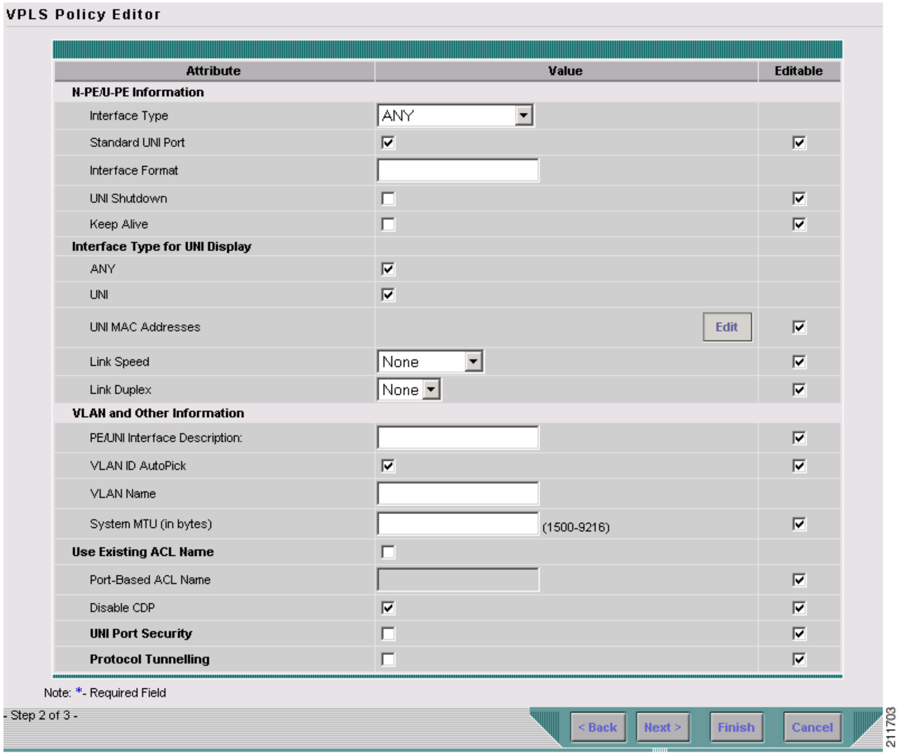
Perform the following steps.

**Step 1** Click **Next**.

The window in Figure 9-34 appears.

The **Editable** check box gives you the option of making a field editable. If you check the **Editable** check box, the service operator who is using this VPLS policy can modify the editable parameter during VPLS service request creation.

*Figure 9-34 Ethernet/EMS (EP-LAN) without CE Policy Attributes*



VPLS Policy Editor

| Attribute | Value | Editable |
|---|---|---|
| **N-PE/U-PE Information** | | |
| Interface Type | ANY | |
| Standard UNI Port | ☑ | ☑ |
| Interface Format | | |
| UNI Shutdown | ☐ | ☑ |
| Keep Alive | ☐ | ☑ |
| **Interface Type for UNI Display** | | |
| ANY | ☑ | |
| UNI | ☑ | |
| UNI MAC Addresses | Edit | ☑ |
| Link Speed | None | ☑ |
| Link Duplex | None | ☑ |
| **VLAN and Other Information** | | |
| PE/UNI Interface Description: | | ☑ |
| VLAN ID AutoPick | ☑ | ☑ |
| VLAN Name | | |
| System MTU (in bytes) | (1500-9216) | ☑ |
| **Use Existing ACL Name** | ☐ | |
| Port-Based ACL Name | | ☑ |
| Disable CDP | ☑ | ☑ |
| **UNI Port Security** | ☐ | ☑ |
| **Protocol Tunnelling** | ☐ | ☑ |

Note: *- Required Field

- Step 2 of 3 -

**Step 2** Choose an **Interface Type** from the drop-down list.

You can choose a particular interface on a CE, N-PE, U-PE, or PE-AGG interface based on the service provider's POP design. The interfaces are:

- **ANY** (Any interface can be chosen.)
- **Port-Channel** (A bundle of ports that share the same characteristics—this gives the service provider the ability to aggregate bandwidth and protection.)
- **Ethernet**
- **FastEthernet**
- **GE-WAN**
- **GigabitEthernet**
- **TenGigabitEthernet**
- **TenGigE**

The value defined here functions as a filter to restrict the interface types an operator can see during VPLS service request creation. If defined as ANY, the operator can see all interface types.

**Step 3** Check the **Standard UNI Port** check box to enable port security.

This is the default. When you uncheck the check box, the port is treated as an uplink with no security features, and the window dynamically changes to eliminate items related to port security.

**Step 4** Enter an **Interface Format** as the slot number/port number for the CE interface (for example, **1/0** indicates that the interface is located at slot 1, port 0).

This is especially useful to specify here if you know that the link will always go through a particular interface’s slot/port location on all or most of the network devices in the service.

**Step 5** Choose a N-PE/U-PE **Encapsulation** type.

The choices are:

- **DOT1Q**
- **DEFAULT**

**Note** When creating a service request based on the Ethernet/EMS (EP-LAN) without CE policy, the Encapsulation attribute is ignored. Therefore, setting this value has no effect.

**Step 6** Check the **UNI Shutdown** check box if you want to leave the UNI port shut during service activation, for example, when the service provider wants to deploy a service in the network but wants to activate it at a later time.

**Step 7** Check the **Keep Alive** check box to configure keepalives on the UNI port.

By default, this check box is unchecked, which causes the command **no keepalive** to be provisioned on the UNI port. This prevents a CPE from sending keepalive packets to the U-PE, for security purposes. This attribute is editable to support modification on a per-service request basis.

**Step 8** Check the **ANY** check box to display all interface types as choices for the UNI interface (when creating service requests based on this policy).

This check box is checked by default.

**Step 9** Check the **UNI** check box to display all interfaces defined as type UNI as choices for the UNI interface (when creating service requests based on this policy).

This check box is checked by default.

**Step 10** Enter one or more Ethernet MAC addresses in **UNI MAC addresses**.

This selection is present only if you uncheck the **Use Existing ACL Name** check box. Click the **Edit** button to bring up a pop-up window in which you enter MAC addresses to be allowed or denied on the port. You can also specify a range of addresses by setting a base MAC address and a filtered MAC address.

**Step 11** Enter a **Link Speed** (optional) of None, 10, 100, 1000, Auto, or nonegotiate.

**Step 12** Enter a **Link Duplex** (optional) of None, Full, Half, or Auto.

**Step 13** In the **PE/UNI Interface Description** field, enter an optional description, for example *Customer-B EMS (EP-LAN) Service*.

**Step 14** Check the **VLAN ID AutoPick** check box if you want ISC to choose a VLAN ID. If you do not check this check box, you will be prompted to provide the VLAN in a Provider VLAN ID field during service activation.

**Step 15** Enter a **VLAN NAME** (optional) to specify a name to describe the VLAN.

The name must be one token (no spaces allowed.) The limit for the VLAN name is 32 characters. The name has to be unique. Two VLANs cannot share the same name.

**Step 16** Enter the **System MTU** in bytes.

The maximum transmission unit (MTU) size is configurable and optional. ISC does not perform an integrity check for this customized value. If a service request goes to the **Failed Deploy** state because this size is not accepted, you must adjust the size until the service request is deployed. ISC supports, ranges for different platforms, as specified below. The range is 1500 to 9216.

- For the 3750 and 3550 platforms, the MTU range is 1500-1546.
- For the 7600 ethernet port, the MTU size is always 9216. Even with the same platform and same IOS release, different line cards support the MTU differently. For example, older line cards only take an MTU size of 9216 and newer cards support 1500-9216. However, ISC uses 9216 in both cases.
- For the 7600 SVI (interface VLAN), the MTU size is 1500-9216.

**Step 17** Check the **Use Existing ACL Name** check box if you want assign your own named access list to the port.

By default, this check box is not checked and ISC automatically assigns a MAC-based ACL on the customer facing UNI port, based on values you enter in **UNI MAC addresses** (below).

**Step 18** Enter a **Port-Based ACL Name** (if you checked the **Use Existing ACL Name** check box, as mentioned in the previous step).

**Note** ISC does not create this ACL automatically. The ACL must already exist on the device, or be added as part of a template, before the service request is deployed. Otherwise, deployment will fail.

**Step 19** Check the **Disable CDP** check box if you want to disable the Cisco Discover Protocol (CDP) on the UNI port.

**Step 20** Check the **UNI Port Security** check box (see Figure 9-35) if you to want to provision port security-related CLIs to the UNI port by controlling the MAC addresses that are allowed to go through the interface.

**a.** For **Maximum Number of MAC address**, enter the number of MAC addresses allowed for port security.

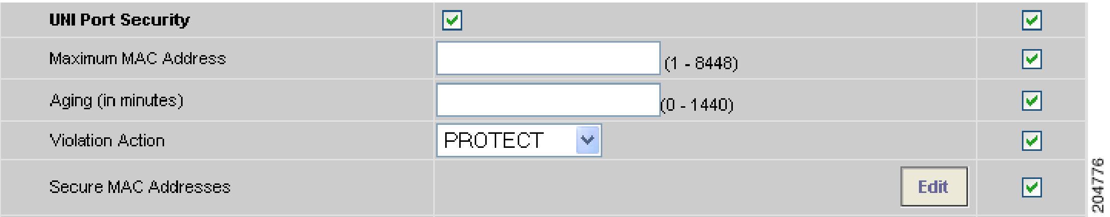
**b.** For **Aging,** enter the length of time the MAC address can stay on the port security table.

**c.** For **Violation Action**, choose what action will occur when a port security violation is detected:

- **PROTECT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value.
- **RESTRICT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value and causes the Security Violation counter to increment.
- **SHUTDOWN**—Puts the interface into the error-disabled state immediately and sends an SNMP trap notification.

**d.** In the **Secure MAC Addresses** field, enter one or more Ethernet MAC addresses.

*Figure 9-35 UNI Port Security*

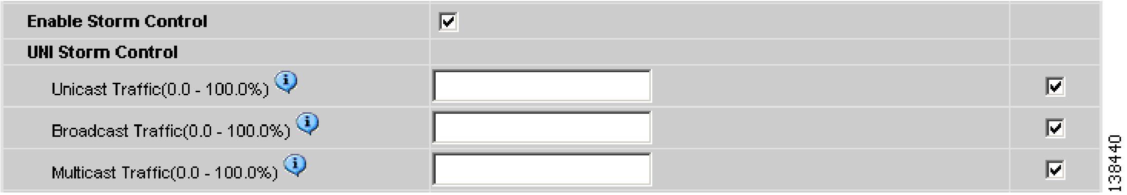


| | | |
|---|---|---|
| **UNI Port Security** | ☑ | ☑ |
| Maximum MAC Address | (1 - 8448) | ☑ |
| Aging (in minutes) | (0 - 1440) | ☑ |
| Violation Action | PROTECT | ☑ |
| Secure MAC Addresses | Edit | ☑ |

**Step 21** Check the **Enable Storm Control** check box (see Figure 9-36) to help prevent the UNI port from being disrupted by a broadcast, multicast or unicast storm.

Enter a threshold value for each type of traffic. The value, which can be specified to two significant digits, represents the percentage of the total available bandwidth of the port. If the threshold of a traffic type is reached, further traffic of that type is suppressed until the incoming traffic falls below the threshold level.

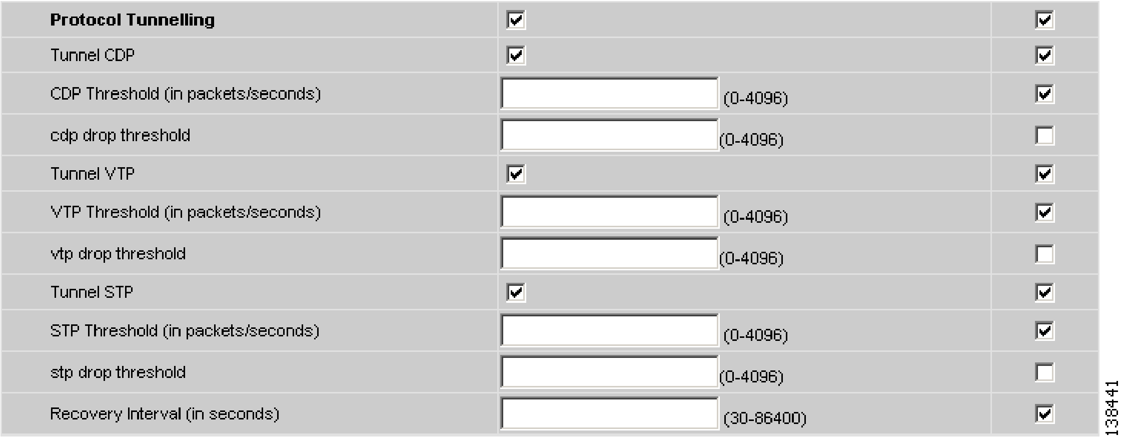
*Figure 9-36 Enable Storm Control*



| | | |
|---|---|---|
| **Enable Storm Control** | ☑ | |
| **UNI Storm Control** | | |
| Unicast Traffic(0.0 - 100.0%) | | ☑ |
| Broadcast Traffic(0.0 - 100.0%) | | ☑ |
| Multicast Traffic(0.0 - 100.0%) | | ☑ |

**Step 22** Check the **Protocol Tunnelling** check box (see Figure 9-37) if you want to define the Layer 2 Bridge Protocol Data Unit (BPDU) frames that can be tunneled over the core to the other end.

*Figure 9-37 Protocol Tunnelling*



| | | |
|---|---|---|
| **Protocol Tunnelling** | ☑ | ☑ |
| Tunnel CDP | ☑ | ☑ |
| CDP Threshold (in packets/seconds) | (0-4096) | ☑ |
| cdp drop threshold | (0-4096) | ☐ |
| Tunnel VTP | ☑ | ☑ |
| VTP Threshold (in packets/seconds) | (0-4096) | ☑ |
| vtp drop threshold | (0-4096) | ☐ |
| Tunnel STP | ☑ | ☑ |
| STP Threshold (in packets/seconds) | (0-4096) | ☑ |
| stp drop threshold | (0-4096) | ☐ |
| Recovery Interval (in seconds) | (30-86400) | ☑ |

For each protocol that you check, enter the shutdown threshold and drop threshold for that protocol:

a. **Tunnel CDP**—Enable Layer 2 tunnelling on Cisco Discover Protocol (CDP).

b. **CDP Threshold—**Enter the number of packets per second to be received before the interface is shut down.

c. **cdp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping CDP packets.

d. **Tunnel VTP**—Enable Layer 2 tunnelling on VLAN Trunk Protocol (VTP).

e. **VTP threshold**—Enter the number of packets per second to be received before the interface is shut down.

f. **vtp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping VTP packets.

g. **Tunnel STP—**Enable Layer 2 tunnelling on Spanning Tree Protocol (STP).

h. **STP Threshold—**Enter the number of packets per second to be received before the interface is shut down.

- i. **stp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping STP packets.
- j. **Recovery Interval**—Enter the amount of time, in seconds, to wait before recovering a UNI port.

**Step 23** Click the **Next** button, if you want to enable template support for the policy.

The Template Association window appears. In this window, you can enable template support and, optionally, associate templates and data files with the policy. For instructions about associating templates with policies and how to use the features in this window, see Appendix B, "Working with Templates and Data Files". When you have completed setting up templates and data files for the policy, click **Finish** in the Template Association window to close it and return to the Policy Editor window.

**Step 24** Click **Finish**.

**Note** The VC ID is mapped from the VPN ID. By default, ISC will "auto pick" this value. However, you can set this manually, if desired. This is done by editing the associated VPN configuration. The Edit VPN window has an **Enable VPLS** check box. When you check this box, you can manually enter a VPN ID in a field provided. For more information on creating and modifying VPNs, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*.

CHAPTER 10

# Managing a VPLS Service Request

This chapter contains the basic steps to provision a VPLS service. It contains the following sections:

## Introducing VPLS Service Requests

A VPLS service request consists of one or more attachment circuits, connecting various sites in a multipoint topology. When you create a service request, you enter several parameters, including the specific interfaces on the CE and PE routers and UNI parameters. You can also integrate a Cisco IP Solution Center (ISC) template with a service request. You can associate one or more templates to the CE, PE, or any U-PE in the middle.

To create a service request, a service policy must already be defined, as described in Chapter 9, “Creating a VPLS Policy.” Based on the predefined VPLS policy, an operator creates a VPLS service request, with or without modifications to the VPLS policy, and deploys the service. The service request must be the same service type (ERMS/EVP-LAN or EMS/EP-LAN) as the policy selected. Service creation and deployment are normally performed by regular network technicians for daily operation of network provisioning.

The following steps are involved in creating a service request for Layer 2 connectivity between customer sites:

- Choose a VPLS policy.
- Choose a VPN. For more information, see Defining VPNs, page 2-4.
- Add a link.
- Choose a CE or UNI interface.
- Choose a Named Physical Circuit (NPC) if more than one NPC exists from the CE or the UNI interface.
- Edit the link attributes.

# Creating a VPLS Service Request

To create a VPLS service request, perform the following steps.

**Step 1** Choose **Service Inventory > Inventory and Connection Manager > Service Requests**.

The Service Requests window appears.

**Step 2** Click **Create**.

**Step 3** Choose **VPLS** from the drop-down list.

VPLS service requests must be associated with a VPLS policy. You choose a VPLS policy from the policies previously created (see Chapter 9, "Creating a VPLS Policy").

**Step 4** If more than one VPLS policy exists, a list of VPLS policies appears. Choose the button for the VPLS policy of choice.

**Step 5** After you make the choice, click **OK**.

The new service request inherits all the properties of that VPLS policy, such as all the editable and noneditable features and preset parameters.

# Creating a VPLS Service Request with a CE

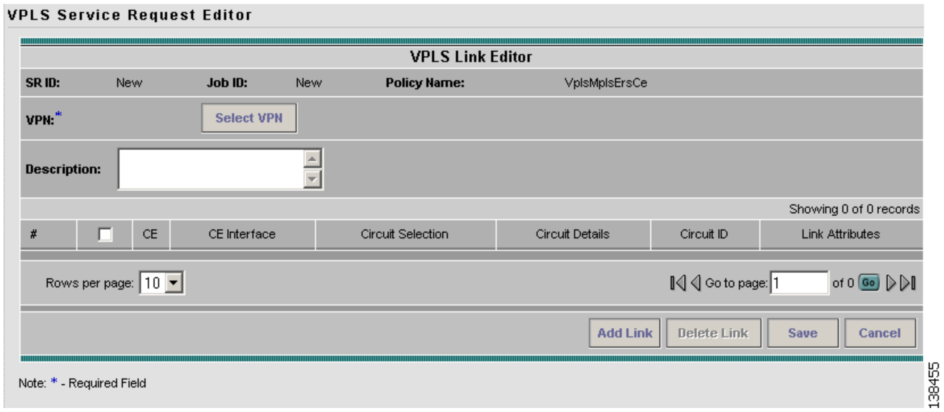
This section includes detailed steps for creating a VPLS service request with a CE present. In this example, the service request is for an VPLS policy over an MPLS core with an ERMS (EVP-LAN) service type and CE present.

Perform the following steps.

**Step 1** Choose the appropriate VPLS policy.

The VPLS Service Request Editor window appears. (See Figure 10-1.)

***Figure 10-1 VPLS Service Request Editor***



VPLS Service Request Editor

VPLS Link Editor

| | | | | | |
|---|---|---|---|---|---|
| SR ID: | New | Job ID: | New | Policy Name: | VplsMplsErsCe |

VPN:*

Description:

Showing 0 of 0 records

| # | ☐ | CE | CE Interface | Circuit Selection | Circuit Details | Circuit ID | Link Attributes |
|---|---|---|---|---|---|---|---|

Rows per page: 10

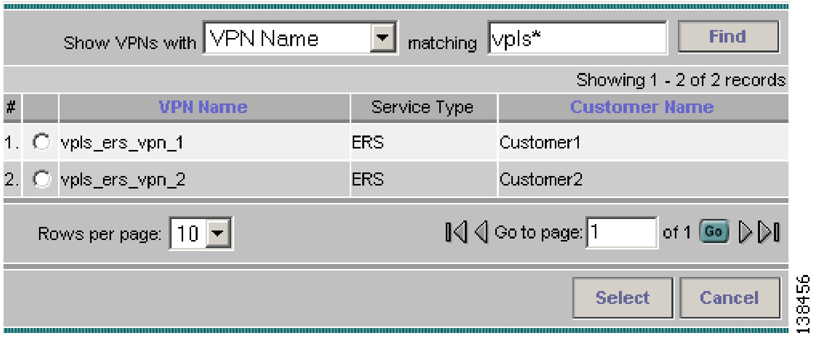
Go to page: 1 of 0

Note: * - Required Field

**Step 2** Click **Select VPN** to choose a VPN for use with this CE.

The Select VPN window appears with the VPNs defined in the system. Only VPNs with the same service type (ERMS/EVP-LAN or EMS/EP-LAN) as the policy you chose appear. (See Figure 10-2.)

*Figure 10-2 Select a VPN*



Show VPNs with: VPN Name matching: vpls*

Showing 1 - 2 of 2 records

| # | | VPN Name | Service Type | Customer Name |
|---|---|---|---|---|
| 1. | ○ | vpls_ers_vpn_1 | ERS | Customer1 |
| 2. | ○ | vpls_ers_vpn_2 | ERS | Customer2 |

Rows per page: 10 Go to page: 1 of 1

**Note** The VC ID is mapped from the VPN ID. By default, ISC will "auto pick" this value. However, you can set this manually, if desired. This is done by editing the associated VPN configuration. The Edit VPN window has an **Enable VPLS** check box. When you check this check box, you can manually enter a VPN ID in a field provided. For more information on creating and modifying VPNs, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*.

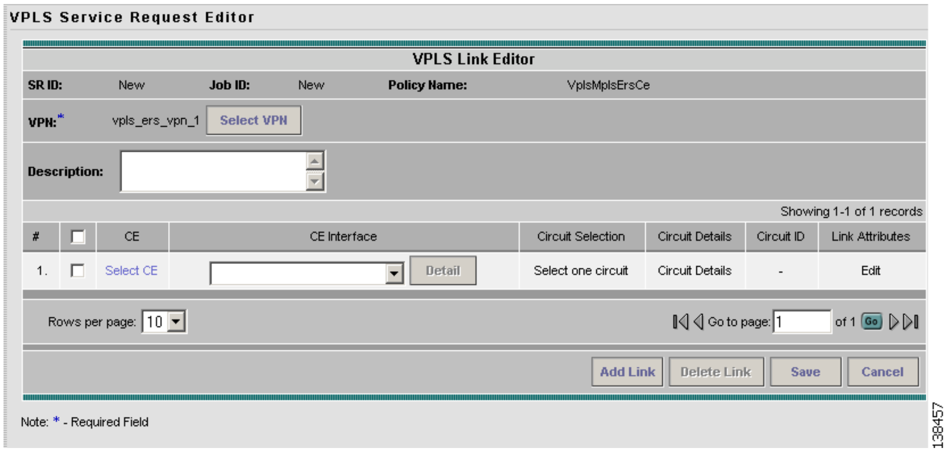
**Step 3** Choose a **VPN Name** in the Select column.

**Step 4** Click **Select**. The VPLS Link Editor window appears with the VPN name displayed.

**Step 5** Click **Add Link**.

You specify the CE endpoints using the VPLS Link Editor. You can add one or more links from a window like the one in Figure 10-3.

*Figure 10-3 Select CE*



VPLS Service Request Editor

VPLS Link Editor

SR ID: New Job ID: New Policy Name: VplsMplsErsCe

VPN:* vpls_ers_vpn_1

Description:

Showing 1-1 of 1 records

| # | □ | CE | CE Interface | Circuit Selection | Circuit Details | Circuit ID | Link Attributes |
|---|---|---|---|---|---|---|---|
| 1. | □ | Select CE | | Select one circuit | Circuit Details | - | Edit |

Rows per page: 10 Go to page: 1 of 1

Note: * - Required Field

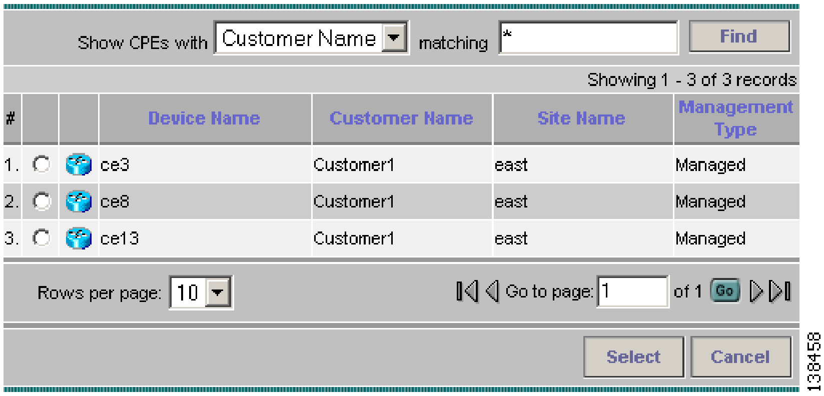
**Step 6** You can enter a description for the service request in the first **Description** field.

The description will show up in this window and also in the Description column of the Service Requests window. The maximum length for this field is 256 characters.

**Step 7** Click **Select CE** in the CE column.

The Select CPE Device window appears. (See Figure 10-4.)

***Figure 10-4 Select CPE Device***



Show CPEs with: Customer Name, matching: *

Showing 1 - 3 of 3 records

| # | | | Device Name | Customer Name | Site Name | Management Type |
|---|---|---|---|---|---|---|
| 1. | ○ | | ce3 | Customer1 | east | Managed |
| 2. | ○ | | ce8 | Customer1 | east | Managed |
| 3. | ○ | | ce13 | Customer1 | east | Managed |

Rows per page: 10; Go to page: 1 of 1

This window displays the list of currently defined CEs.

a. From the **Show CPEs with** drop-down list, you can display CEs by Customer Name, by Site, or by Device Name.

b. You can use the **Find** button to either search for a specific CE, or to refresh the display.

c. You can set the **Rows per page** to 5, 10, 20, 30, 40, or All.

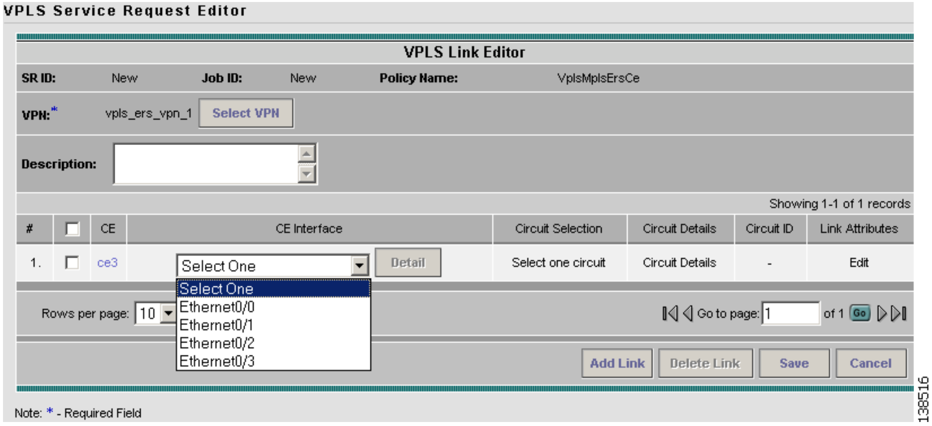
**Step 8** In the Select column, choose a CE for the VPLS link.

**Step 9** Click **Select**.

The VPLS Link Editor window appears displaying the name of the selected CE in the CE column.

**Step 10** Choose the CE interface from the drop-down list. (See Figure 10-5.)

***Figure 10-5 Select the CE Interface***



VPLS Service Request Editor

VPLS Link Editor

SR ID: New Job ID: New Policy Name: VplsMplsErsCe

VPN:* vpls_ers_vpn_1

Description:

Showing 1-1 of 1 records

| # | □ | CE | CE Interface | | Circuit Selection | Circuit Details | Circuit ID | Link Attributes |
|---|---|---|---|---|---|---|---|---|
| 1. | □ | ce3 | Select One | Detail | Select one circuit | Circuit Details | - | Edit |

Rows per page: 10; Go to page: 1 of 1

Note: * - Required Field

**Note** When you provision an ERMS (EVP-LAN) service (and when you choose a UNI for a particular device), ISC determines if there are other services using the same UNI. If so, a warning message is displayed. If you ignore the message and save the service request, all of the underlying service requests lying on the same UNI are synchronized with the modified shared attributes of the latest service request. In addition, the state of the existing service requests is changed to the Requested state.

**Step 11** Click **Select one circuit** in the Circuit Selection column.

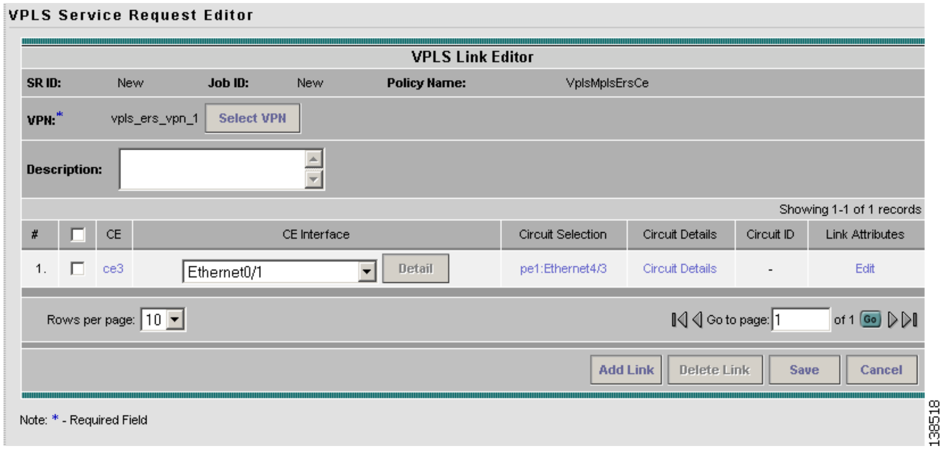
The Select NPC window appears. If only one NPC exists for the chosen CE and CE interface, that NPC is automatically populated in the Circuit Selection column and you need not choose it explicitly.

**Step 12** Choose the name of the NPC from the Select column.

**Step 13** Click **OK**.

Each time you choose a CE and its interface, the NPC that was precreated from this CE and interface is automatically displayed under **Circuit Selection**. (See Figure 10-6.) This means that you do not have to further specify the PE to complete the link.

***Figure 10-6 NPC Selected***



VPLS Service Request Editor

VPLS Link Editor

SR ID: New Job ID: New Policy Name: VplsMplsErsCe

VPN:* vpls_ers_vpn_1

Description:

Showing 1-1 of 1 records

| # | □ | CE | CE Interface | Circuit Selection | Circuit Details | Circuit ID | Link Attributes |
|---|---|---|---|---|---|---|---|
| 1. | □ | ce3 | Ethernet0/1 | pe1:Ethernet4/3 | Circuit Details | - | Edit |

Rows per page: 10

Go to page: 1 of 1

Note: * - Required Field

**Step 14** If you want to review the details of this NPC, click **Circuit Details** in the Circuit Details column.

The NPC Details window appears and lists the circuit details for this NPC.

**Step 15** The Circuit ID is created automatically, based on the VLAN data for the circuit.

**Step 16** To edit values that were set by the VPLS policy, that is, the values that were marked “editable” during the VPLS policy creation, click the **Edit** link in the Link Attributes column for a link.

The Link Attributes window appears.

**Note** For more information on setting attributes in this window, see Modifying the VPLS Service Request, page 10-10.

**Note** For information on the Bridge Domain ID attribute, which shows up in some VPLS service request scenarios, see Modifying the VPLS Service Request, page 10-10.

**Step 17** Continue to specify additional CEs, as in previous steps, if desired.

**Step 18** Click **OK**.

**Step 19** Click **Save**.

The service request is created and saved into ISC.

# Creating a VPLS Service Request without a CE

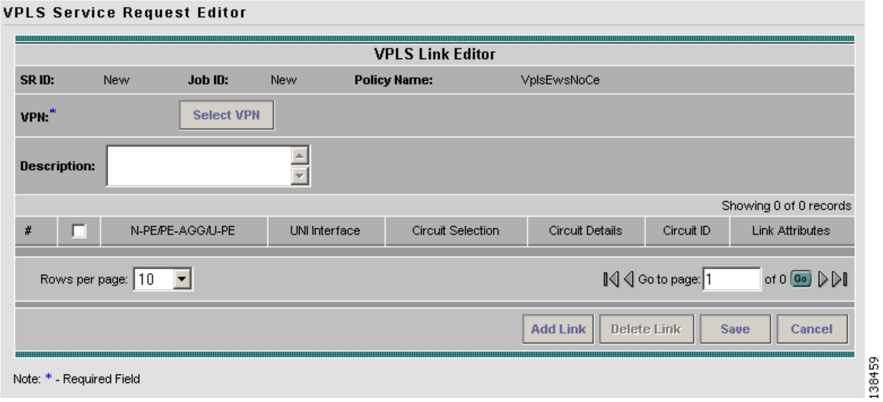
This section includes detailed steps for creating a VPLS service request without a CE present. In this example, the service request is for an VPLS policy over an MPLS core with an EMS (EP-LAN) service type and no CE present.

Perform the following steps.

**Step 1** Choose the appropriate VPLS policy.

The VPLS Service Request Editor window appears. (See Figure 10-7.)

***Figure 10-7 VPLS Service Request Editor***



VPLS Service Request Editor

VPLS Link Editor

SR ID: New Job ID: New Policy Name: VplsEwsNoCe

VPN:* Select VPN

Description:

Showing 0 of 0 records

| # | □ | N-PE/PE-AGG/U-PE | UNI Interface | Circuit Selection | Circuit Details | Circuit ID | Link Attributes |
|---|---|---|---|---|---|---|---|

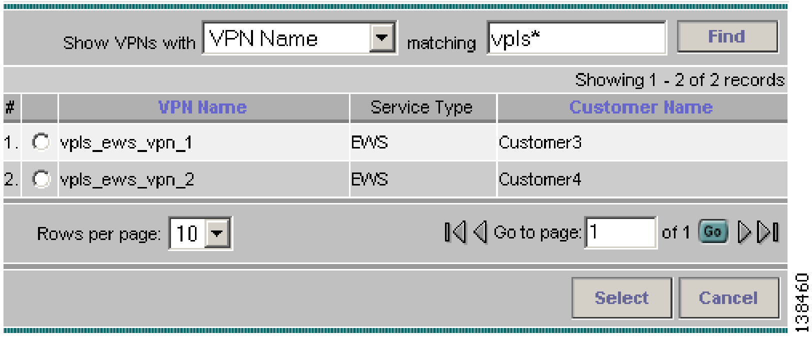
Rows per page: 10 Go to page: 1 of 0

Note: * - Required Field

**Step 2** Click **Select VPN** to choose a VPN for use with this PE.

The Select VPN window appears with the VPNs defined in the system. Only VPNs with the same service type (ERMS/EVP-LAN or EMS/EP-LAN) as the policy you chose appear. (See Figure 10-8.)

*Figure 10-8* *Select a VPN*



Show VPNs with VPN Name matching vpls* Find

Showing 1 - 2 of 2 records

| # | | VPN Name | Service Type | Customer Name |
|---|---|---|---|---|
| 1. | ○ | vpls_ews_vpn_1 | EWS | Customer3 |
| 2. | ○ | vpls_ews_vpn_2 | EWS | Customer4 |

Rows per page: 10 Go to page: 1 of 1

**Note** The VC ID is mapped from the VPN ID. By default, ISC will “auto pick” this value. However, you can set this manually, if desired. This is done by editing the associated VPN configuration. The Edit VPN window has an **Enable VPLS** check box. When you check this check box, you can manually enter a VPN ID in a field provided. For more information on creating and modifying VPNs, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*.

**Step 3** Choose a **VPN Name** in the Select column.

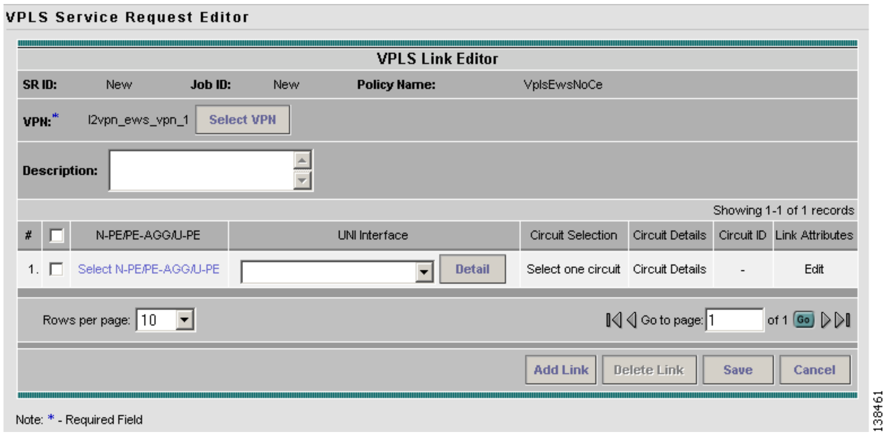
**Step 4** Click **Select**.

The VPLS Link Editor window appears with the VPN name displayed.

**Step 5** Click **Add Link**.

You specify the U-PE/PE-AGG/U-PE endpoints using the VPLS Link Editor. You can add one or more links from a window like the one in Figure 10-9.

*Figure 10-9* *Select N-PE/PE-AGG/U-PE*



VPLS Service Request Editor

VPLS Link Editor

SR ID: New Job ID: New Policy Name: VplsEwsNoCe

VPN:* l2vpn_ews_vpn_1

Description:

Showing 1-1 of 1 records

| # | □ | N-PE/PE-AGG/U-PE | UNI Interface | Circuit Selection | Circuit Details | Circuit ID | Link Attributes |
|---|---|---|---|---|---|---|---|
| 1. | □ | Select N-PE/PE-AGG/U-PE | Detail | Select one circuit | Circuit Details | - | Edit |

Rows per page: 10 Go to page: 1 of 1

Note: * - Required Field

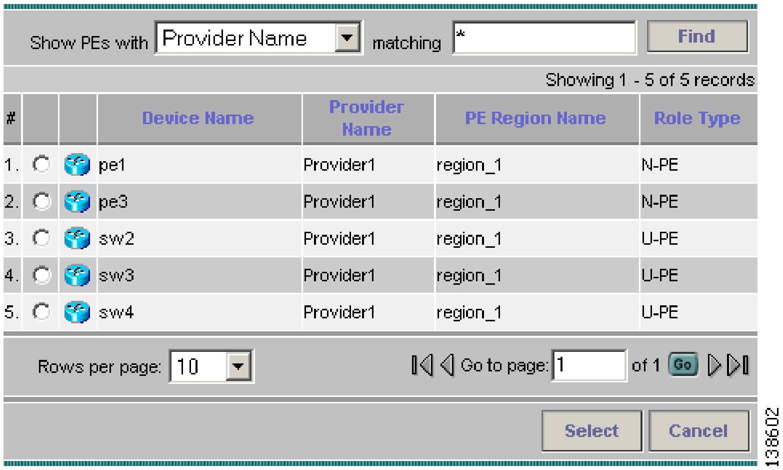
**Step 6** You can enter a description for the service request in the first **Description** field.

The description will show up in this window and also in the Description column of the Service Requests window. The maximum length for this field is 256 characters.

**Step 7** Click **Select N-PE/PE-AGG/U-PE** in the N-PE/PE-AGG/U-PE column.

The Select PE Device window appears. (See Figure 10-10.)

*Figure 10-10 Select PE Device*



Show PEs with Provider Name matching * Find

Showing 1 - 5 of 5 records

| # | | | Device Name | Provider Name | PE Region Name | Role Type |
|---|---|---|---|---|---|---|
| 1. | ○ | | pe1 | Provider1 | region_1 | N-PE |
| 2. | ○ | | pe3 | Provider1 | region_1 | N-PE |
| 3. | ○ | | sw2 | Provider1 | region_1 | U-PE |
| 4. | ○ | | sw3 | Provider1 | region_1 | U-PE |
| 5. | ○ | | sw4 | Provider1 | region_1 | U-PE |

Rows per page: 10 Go to page: 1 of 1 Go

This window displays the list of currently defined PEs.

a. The **Show PEs with** drop-down list shows PEs by customer name, by site, or by device name.

b. The **Find** button allows a search for a specific PE or a refresh of the window.

c. The **Rows per page** drop-down list allows the page to be set to 5, 10, 20, 30, 40, or All.

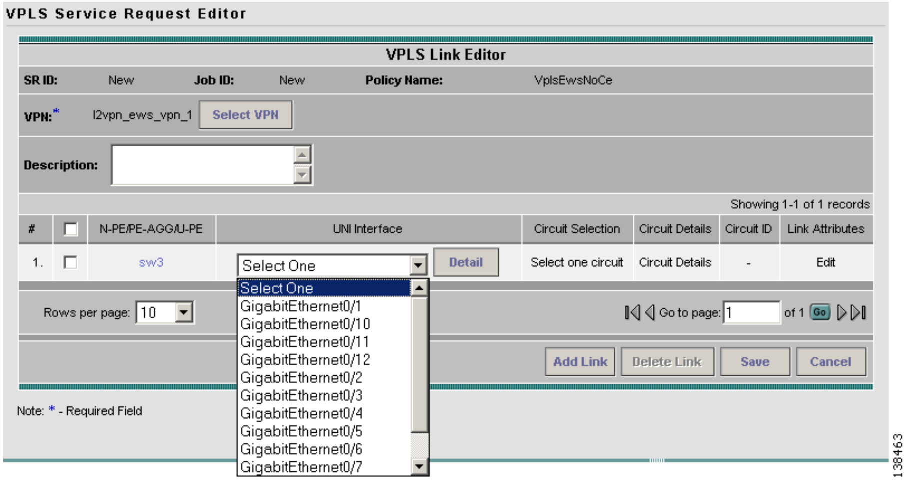
**Step 8** In the **Select** column, choose the PE device name for the VPLS link.

**Step 9** Click **Select**.

The VPLS Link Editor window appears displaying the name of the selected N-PE/PE-AGG/U-PE in the N-PE/PE-AGG/U-PE column

**Step 10** Choose the UNI interface from the drop-down list. (See Figure 10-11.)

*Figure 10-11 Select the UNI Interface*



VPLS Service Request Editor

VPLS Link Editor

SR ID: New Job ID: New Policy Name: VplsEwsNoCe

VPN:* l2vpn_ews_vpn_1 Select VPN

Description:

Showing 1-1 of 1 records

| # | □ | N-PE/PE-AGG/U-PE | UNI Interface | | Circuit Selection | Circuit Details | Circuit ID | Link Attributes |
|---|---|---|---|---|---|---|---|---|
| 1. | □ | sw3 | Select One | Detail | Select one circuit | Circuit Details | - | Edit |

Select One
GigabitEthernet0/1
GigabitEthernet0/10
GigabitEthernet0/11
GigabitEthernet0/12
GigabitEthernet0/2
GigabitEthernet0/3
GigabitEthernet0/4
GigabitEthernet0/5
GigabitEthernet0/6
GigabitEthernet0/7

Rows per page: 10 Go to page: 1 of 1 Go

Note: * - Required Field

**Note** When you provision an ERMS (EVP-LAN) service (and when you choose a UNI for a particular device), ISC determines if there are other services using the same UNI. If so, a warning message is displayed. If you ignore the message and save the service request, all of the underlying service requests lying on the same UNI are synchronized with the modified shared attributes of the latest service request. In addition, the state of the existing service requests is changed to the Requested state.

**Step 11** If the PE role type is U-PE, click **Select one circuit** in the Circuit Selection column.

The Select NPC window appears. If only one NPC exists for the chosen PE and PE interface, that NPC is automatically populated in the Circuit Selection column and you need not choose it explicitly.

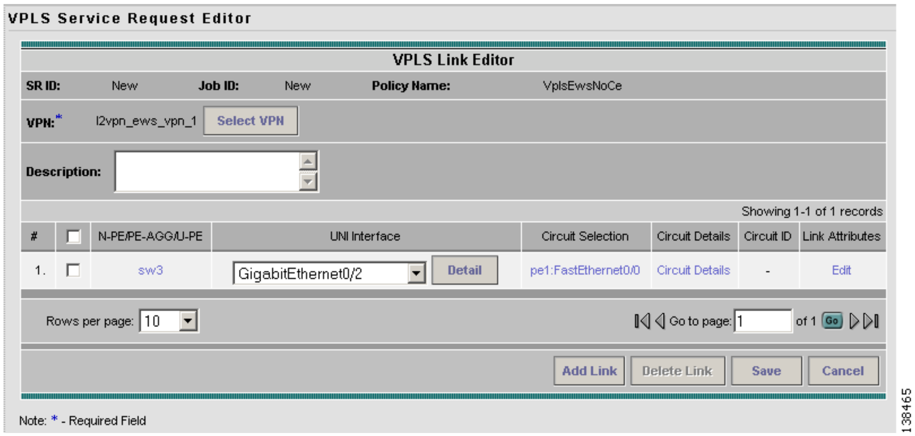
**Note** If the PE role type is N-PE, the columns Circuit Selection and Circuit Details are disabled.

**Step 12** Choose the name of the NPC from the **Select** column.

**Step 13** Click **OK**.

Each time you choose a PE and its interface, the NPC that was precreated from this PE and interface is automatically displayed under **Circuit Selection**. (See Figure 10-12.) This means that you do not have to further specify the PE to complete the link.

*Figure 10-12 NPC Created*



VPLS Service Request Editor

VPLS Link Editor

SR ID: New Job ID: New Policy Name: VplsEwsNoCe

VPN:* l2vpn_ews_vpn_1 Select VPN

Description:

Showing 1-1 of 1 records

| # | □ | N-PE/PE-AGG/U-PE | UNI Interface | Circuit Selection | Circuit Details | Circuit ID | Link Attributes |
|---|---|---|---|---|---|---|---|
| 1. | □ | sw3 | GigabitEthernet0/2 Detail | pe1:FastEthernet0/0 | Circuit Details | - | Edit |

Rows per page: 10 Go to page: 1 of 1

Note: * - Required Field

**Step 14** If you want to review the details of this NPC, click **Circuit Details** in the Circuit Details column.

The NPC Details window appears and lists the circuit details for this NPC.

The Circuit ID is created automatically, based on the VLAN data for the circuit.

**Step 15** To edit values that were set by the VPLS policy, that is, the values that were marked "editable" during the VPLS policy creation, click the **Edit** link in the Link Attributes column for a link.

**Note** For more information on setting attributes in this window, see Modifying the VPLS Service Request, page 10-10.

**Note** For information on the Bridge Domain ID attribute, which shows up in some VPLS service request scenarios, see Modifying the VPLS Service Request, page 10-10.

**Step 16** Continue to specify additional PEs, as in previous steps, if desired.

**Step 17** Click **Save**.

The VPLS service request is created and saved into ISC.

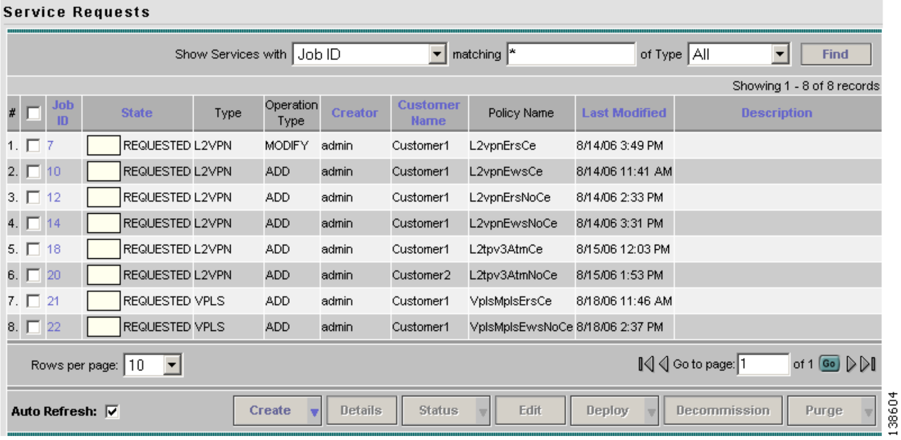
# Modifying the VPLS Service Request

You can modify a VPLS service request if you must change or modify the VPLS links. This is also where you can associate templates and data files to a link.

Perform the following steps.

**Step 1** Choose **Service Inventory** > **Inventory and Connection Manager** > **Service Requests**. (See Figure 10-13.)

*Figure 10-13 VPLS Service Activation*



Service Requests

Show Services with Job ID matching * of Type All

Showing 1 - 8 of 8 records

| # | □ | Job ID | State | Type | Operation Type | Creator | Customer Name | Policy Name | Last Modified | Description |
|---|---|---|---|---|---|---|---|---|---|---|
| 1. | □ | 7 | REQUESTED | L2VPN | MODIFY | admin | Customer1 | L2vpnErsCe | 8/14/06 3:49 PM | |
| 2. | □ | 10 | REQUESTED | L2VPN | ADD | admin | Customer1 | L2vpnEwsCe | 8/14/06 11:41 AM | |
| 3. | □ | 12 | REQUESTED | L2VPN | ADD | admin | Customer1 | L2vpnErsNoCe | 8/14/06 2:33 PM | |
| 4. | □ | 14 | REQUESTED | L2VPN | ADD | admin | Customer1 | L2vpnEwsNoCe | 8/14/06 3:31 PM | |
| 5. | □ | 18 | REQUESTED | L2VPN | ADD | admin | Customer1 | L2tpv3AtmCe | 8/15/06 12:03 PM | |
| 6. | □ | 20 | REQUESTED | L2VPN | ADD | admin | Customer2 | L2tpv3AtmNoCe | 8/15/06 1:53 PM | |
| 7. | □ | 21 | REQUESTED | VPLS | ADD | admin | Customer1 | VplsMplsErsCe | 8/18/06 11:46 AM | |
| 8. | □ | 22 | REQUESTED | VPLS | ADD | admin | Customer1 | VplsMplsEwsNoCe | 8/18/06 2:37 PM | |

Rows per page: 10

Go to page: 1 of 1

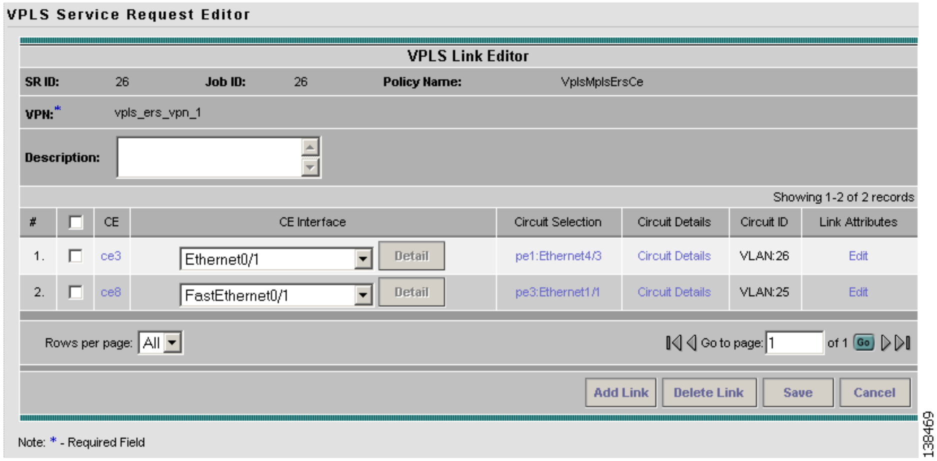
Auto Refresh: ☑

**Step 2** Check a check box for a service request.

**Step 3** Click **Edit**.

The VPLS Link Editor window appears. (See Figure 10-14.)

*Figure 10-14 VPLS Link Editor*



VPLS Service Request Editor

VPLS Link Editor

SR ID: 26 Job ID: 26 Policy Name: VplsMplsErsCe

VPN:* vpls_ers_vpn_1

Description:

Showing 1-2 of 2 records

| # | □ | CE | CE Interface | | Circuit Selection | Circuit Details | Circuit ID | Link Attributes |
|---|---|---|---|---|---|---|---|---|
| 1. | □ | ce3 | Ethernet0/1 | Detail | pe1:Ethernet4/3 | Circuit Details | VLAN:26 | Edit |
| 2. | □ | ce8 | FastEthernet0/1 | Detail | pe3:Ethernet1/1 | Circuit Details | VLAN:25 | Edit |

Rows per page: All

Go to page: 1 of 1

Note: * - Required Field

**Step 4** Specify remaining items in the End-to-End-Wire Editor window, as necessary for your configuration:

- Choose any of the blue highlighted values to edit the VPLS links.
- Click **Add Link** to add a VPLS link.
- Click **Delete Link** to delete a VPLS link.

**Note** If you are attempting to decommission a service request to which a template has been added, see Monitoring Service Requests, page 11-10 for information on the proper way to do this.

- You can enter a description for the service request in the first **Description** field. The description will show up in this window and also in the Description column of the Service Requests window. The maximum length for this field is 256 characters.
- The Circuit ID is created automatically, based on the VLAN data for the circuit.

**Step 5** To modify the link attributes, click **Edit** in the Link Attributes column as shown in the VPLS link editor.

The Link Attributes window appears.

**Step 6** Edit the link attributes as desired.

**Note** If you did not choose **VLANI D AutoPick** in the VPLS policy, you are prompted to provide the VLAN in a **Provider VLAN ID** field.

**Note** For information on the Bridge Domain ID attribute, which shows up in some VPLS service request scenarios, see Modifying the VPLS Service Request, page 10-10.

**Step 7** To add a template and data file to a link, choose a Device Name, and click **Add** under Templates.

The Add/Remove Templates window appears.

**Note** To add a template to a link, you must have already created the template. For detailed steps to create templates, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*. For more information on how to use templates and data files in service requests, see Appendix B, "Working with Templates and Data Files."

**Step 8** Click **Add**.

The Template Data File Chooser window appears.

**Step 9** In the left pane, navigate to and select a template.

The associated data files are listed in rows in the main window.

**Step 10** Check the data file that you want to add and click **Accept**.

The Add/Remove Templates window appears with the template displayed.

**Step 11** Choose a Template name.

**Step 12** Under Action, use the drop-down list and choose **APPEND** or **PREPEND**.

Append tells ISC to append the template generated CLI to the regular ISC (non-template) CLI. Prepend is the reverse and does not append the template to the ISC CLI.

**Step 13** Choose Active to use this template for this service request.

If you do not choose Active, the template is not used.

**Step 14** Click **OK**.

The Link Attributes with the template added appears.

**Step 15** Click **OK**.

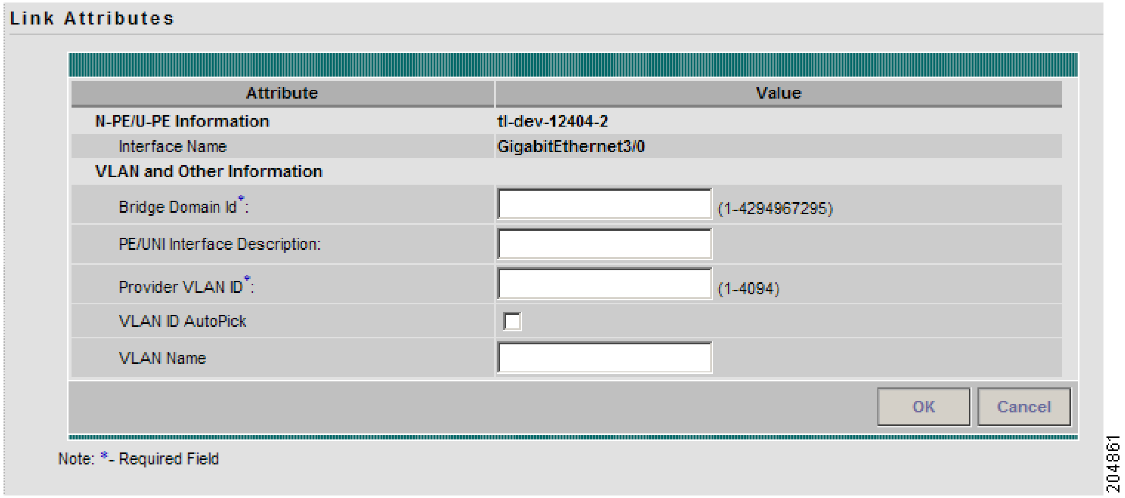
The Service Request Editor window appears.

**Step 16** When you are finished editing the VPLS links, click **Save**.

# Using the Bridge Domain ID Attribute

The Bridge Domain ID attribute appears in the Link Attributes window of some VPLS service request scenarios, as shown in Figure 10-15.

*Figure 10-15 Bridge Domain ID Attribute*



Link Attributes

| Attribute | Value |
|---|---|
| N-PE/U-PE Information | tl-dev-12404-2 |
| Interface Name | GigabitEthernet3/0 |
| VLAN and Other Information | |
| Bridge Domain Id*: | (1-4294967295) |
| PE/UNI Interface Description: | |
| Provider VLAN ID*: | (1-4094) |
| VLAN ID AutoPick | □ |
| VLAN Name | |

Note: *- Required Field

To use the Bridge Domain ID attribute, enter an ID number in the **Bridge Domain ID** text field to enable bridge domain functionality for the VPLS service request.

Acceptable values are 1 to 4294967295.

Usage notes:

- The Bridge Domain ID attribute is only available for the following service request scenarios:
  - Ethernet/ERMS (EVP-LAN) with a CE
  - Ethernet/ERMS (EVP-LAN) without a CE
  - Ethernet/EMS (EP-LAN) with a CE
  - Ethernet/EMS (EP-LAN) without a CE
- The Bridge Domain ID attribute is only supported for the Cisco GSR 12406 running IOS 12.0(32)SY6 and functioning in an N-PE role. This attribute will show up in a service request only for this platform; otherwise, the attribute will be filtered from the Link Attributes window of the service request.
- The following points apply to service requests based on this policy:
  - When an N-PE (GSR platform) is used as a UNI device, the standard UNI attributes are not displayed in the Link Attributes window of the service request workflow.
  - When a U-PE (non-GSR platform) is used as a UNI device, all standard UNI attributes are displayed in the Link Attributes window of the service request workflow.
  - For VPLS EMS services, a U-PE (non-GSR platform) should be used in the same circuit which is terminating on a GSR device (N-PE). In other words, an NPC circuit should be used to provision VPLS EMS on GSR devices.

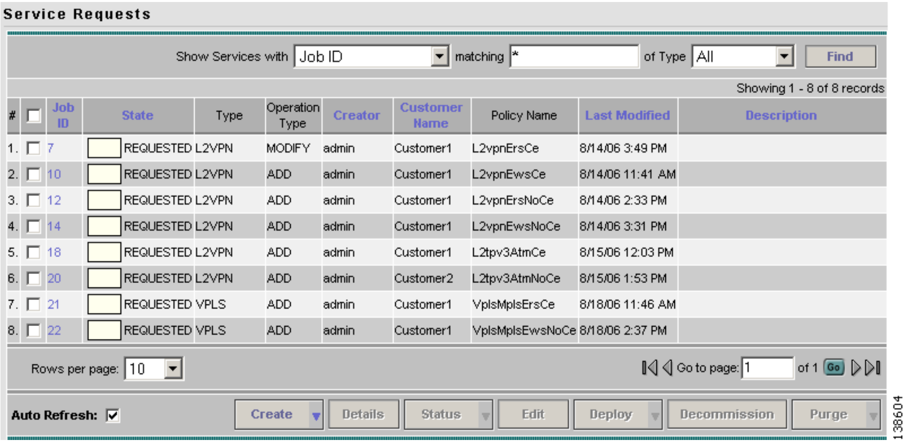
# Saving the VPLS Service Request

To save a VPLS service request, perform the following steps.

**Step 1** When you are finished with Link Attributes for all the Attachment Circuits, click **Save** to finish the VPLS service request creation.

If the VPLS service request is successfully created, you will see the service request list window. (See Figure 10-16.) The newly created VPLS service request is added with the state of REQUESTED, as shown in the figure.

***Figure 10-16 VPLS Service Request Created***



Service Requests

Show Services with: Job ID matching: * of Type: All

Showing 1 - 8 of 8 records

| # | □ | Job ID | State | Type | Operation Type | Creator | Customer Name | Policy Name | Last Modified | Description |
|---|---|---|---|---|---|---|---|---|---|---|
| 1. | □ | 7 | REQUESTED | L2VPN | MODIFY | admin | Customer1 | L2vpnErsCe | 8/14/06 3:49 PM | |
| 2. | □ | 10 | REQUESTED | L2VPN | ADD | admin | Customer1 | L2vpnEwsCe | 8/14/06 11:41 AM | |
| 3. | □ | 12 | REQUESTED | L2VPN | ADD | admin | Customer1 | L2vpnErsNoCe | 8/14/06 2:33 PM | |
| 4. | □ | 14 | REQUESTED | L2VPN | ADD | admin | Customer1 | L2vpnEwsNoCe | 8/14/06 3:31 PM | |
| 5. | □ | 18 | REQUESTED | L2VPN | ADD | admin | Customer1 | L2tpv3AtmCe | 8/15/06 12:03 PM | |
| 6. | □ | 20 | REQUESTED | L2VPN | ADD | admin | Customer2 | L2tpv3AtmNoCe | 8/15/06 1:53 PM | |
| 7. | □ | 21 | REQUESTED | VPLS | ADD | admin | Customer1 | VplsMplsErsCe | 8/18/06 11:46 AM | |
| 8. | □ | 22 | REQUESTED | VPLS | ADD | admin | Customer1 | VplsMplsEwsNoCe | 8/18/06 2:37 PM | |

Rows per page: 10

Go to page: 1 of 1

Auto Refresh: ☑

**Step 2** If, however, the VPLS service request creation failed for some reason (for example, a value chosen is out of bounds), you are warned with an error message.

In such a case, you should correct the error and save the service request again.

CHAPTER 11

# Deploying, Monitoring, and Auditing Service Requests

This chapter describes how to deploy, monitor and audit L2VPN, VPLS or FlexUNI/EVC service requests, and how to access task logs. It contains the following sections:

# Deploying Service Requests

To apply L2VPN, VPLS, or FlexUNI policies to network devices, you must deploy the service request. When you deploy a service request, ISC compares the device information in the Repository (the ISC database) with the current device configuration and generates a configlet.

## Pre-Deployment Changes

You can change the Dynamic Component Properties Library (DCPL) parameter **actionTakenOnUNIVlanList** before you deploy an L2VPN or VPLS service request. This will be necessary if the **trunk allowed vlan** list is not present on the User Network Interface (UNI).

To make this change, perform the following steps.

**Step 1** Choose **Administration > Control Center**.

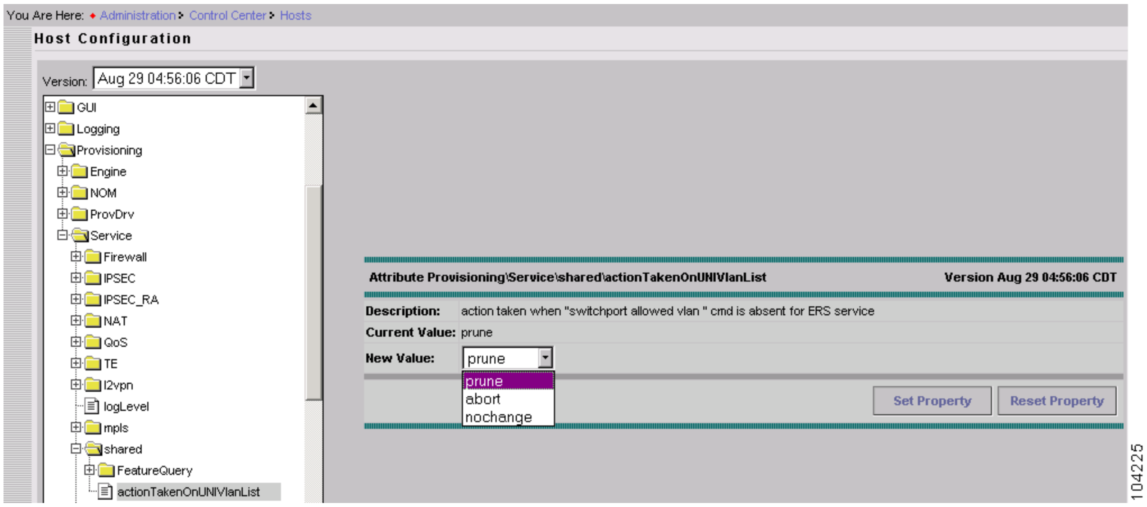
**Step 2** Choose the host that you want to change.

**Step 3** Click **Config**.

**Step 4** Choose **Provisioning > Service > shared > actionTakenOnUNIVlanList**.

The window shown in Figure 11-1 appears.

*Figure 11-1 Change DCPL Parameter*



You Are Here: • Administration > Control Center > Hosts

Host Configuration

Version: Aug 29 04:56:06 CDT

Attribute Provisioning\Service\shared\actionTakenOnUNIVlanList Version Aug 29 04:56:06 CDT

Description: action taken when "switchport allowed vlan " cmd is absent for ERS service

Current Value: prune

New Value: prune

**Step 5** Choose one of the following:

- **prune** to have ISC create the minimum VLAN list. This is the default.
- **abort** to have ISC stop the L2VPN or VPLS service request provisioning with the error message: **trunk allowed vlan list is absent on ERS UNI**.
- **nochange** to have ISC allow all VLANs.

**Step 6** Click **Set Property**.

## Service Deployment

After you create a service request and save it in the ISC repository, you can deploy or force-deploy it.

Perform the following steps.

**Step 1** Choose **Service Inventory > Inventory and Connection Manager > Service Requests**.

The Service Requests window appears.

**Step 2** Choose a service request.

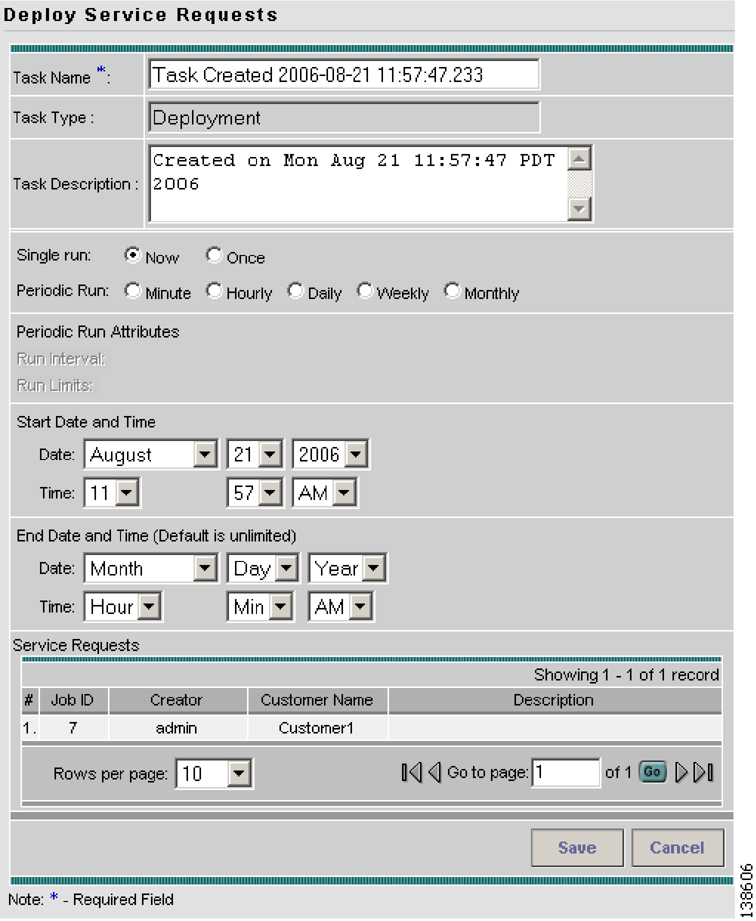
**Step 3** Click **Deploy** and choose **Deploy** or **Force-Deploy**.

Use **Deploy** when the service request state is Requested or Invalid.

Use **Force Deploy** when the service request state is Deployed, Failed Deployed, or Failed Audit.

The Deploy Service Requests window appears. (See Figure 11-2.)

*Figure 11-2 Schedule Service Activation*



Deploy Service Requests

Task Name *: Task Created 2006-08-21 11:57:47.233

Task Type : Deployment

Task Description : Created on Mon Aug 21 11:57:47 PDT 2006

Single run: ● Now ○ Once

Periodic Run: ○ Minute ○ Hourly ○ Daily ○ Weekly ○ Monthly

Periodic Run Attributes

Run Interval:

Run Limits:

Start Date and Time

Date: August 21 2006

Time: 11 57 AM

End Date and Time (Default is unlimited)

Date: Month Day Year

Time: Hour Min AM

Service Requests

Showing 1 - 1 of 1 record

| # | Job ID | Creator | Customer Name | Description |
|---|---|---|---|---|
| 1. | 7 | admin | Customer1 | |

Rows per page: 10 Go to page: 1 of 1

Note: * - Required Field

**Step 4** Choose a schedule for the activation of the service.

**Step 5** After you schedule the service request, click **Save**.

After you schedule the service request, you can monitor the service request that is being deployed. See Verifying Service Requests, page 11-3 and Monitoring Service Requests, page 11-10 for more information.

# Verifying Service Requests

After you deploy a service request, you should verify that there were no errors.

You can verify a service request through the following:

- Transition state—The transition state of a service request is listed on the Service Requests window in the State column. When the service request is successfully deployed, its state changes to DEPLOYED. For more information, see Service Request States, page 11-4.

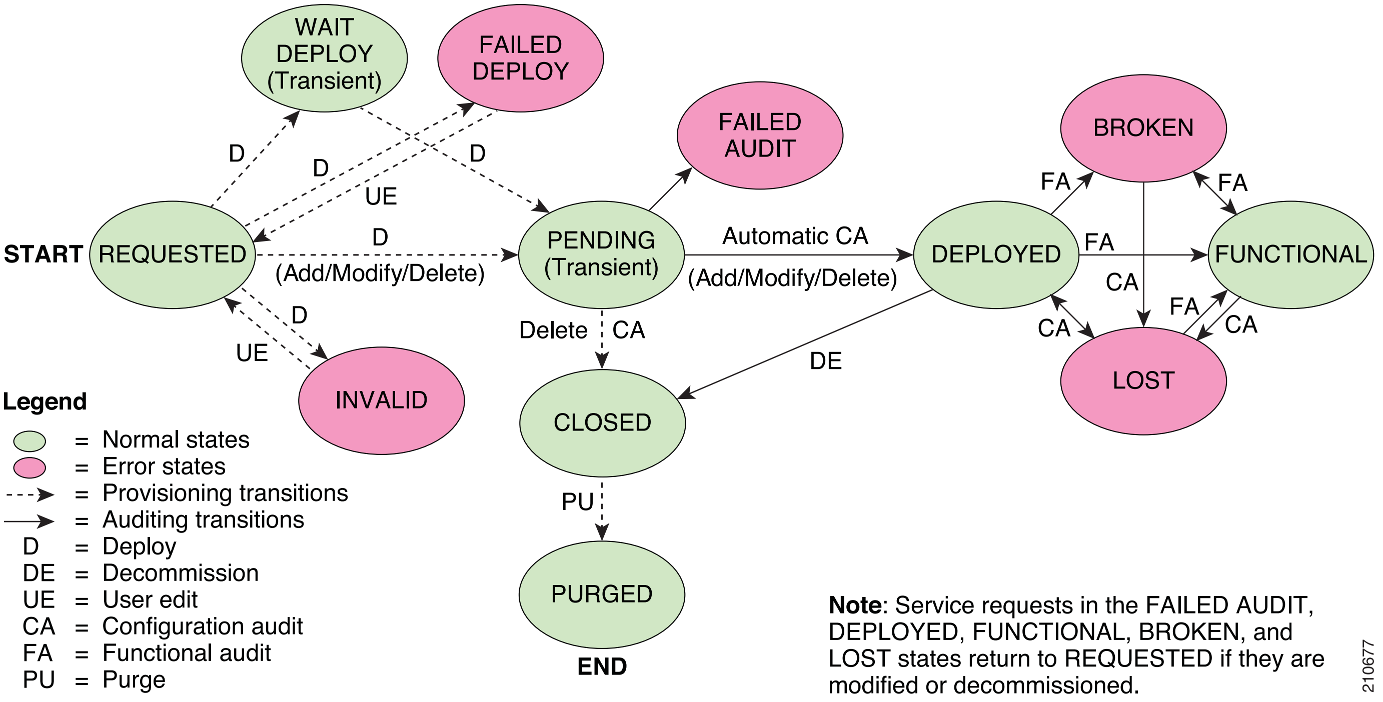
- View service request details—From the Service Requests Details window, you can view the link endpoints and the configlets for this service request. For more information, See Viewing Service Request Details, page 11-7.
- Task Logs—Access the task logs from the Monitoring tab to help you troubleshoot a failed service request or to view more details about a service request. For more information, see Monitoring Service Requests, page 11-10.

# Service Request States

A service request transition state describes the different stages a service request enters during the provisioning process. For example, when you deploy a service request, ISC compares the device information in the Repository (the ISC database) with the current device configuration and generates a configlet. When the configlet is generated and downloaded to the device, the service request enters the **Pending** state. When the device is audited, the service request enters the **Deployed** state.

Figure 11-3, "Service Requests States Transition Diagram," shows a high-level diagram of the relationships and movement among ISC service request states.

***Figure 11-3 Service Requests States Transition Diagram***



**Note**: Service requests in the FAILED AUDIT, DEPLOYED, FUNCTIONAL, BROKEN, and LOST states return to REQUESTED if they are modified or decommissioned.

Table 11-1, "Summary of Cisco IP Solution Center Service Request States," describes the functions of each ISC service request state. They are listed in alphabetic order.

*Table 11-1 Summary of Cisco IP Solution Center Service Request States*

| Service Request Type | Description |
|---|---|
| **Broken**<br>(valid only for MPLS services) | The router is correctly configured but the service is unavailable (due to a broken cable or Layer 2 problem, for example).<br>An MPLS service request moves to **Broken** if the auditor finds the routing and forwarding tables for this service, but they do not match the service intent. |
| **Closed** | A service request moves to **Closed** if the service request should no longer be used during the provisioning or auditing process. A service request moves to the **Closed** state only upon successful audit of a decommission service request. ISC does not remove a service request from the database to allow for extended auditing. Only a specific administrator purge action results in service requests being removed. |
| **Deployed** | A service request moves to **Deployed** if the intention of the service request is found in the router configuration file. **Deployed** indicates that the configuration file has been downloaded to the router, and the intent of the request has been verified at the configuration level. That is, ISC downloaded the configlets to the routers and the service request passed the audit process. |
| **Failed Audit** | This state indicates that ISC downloaded the configlet to the router successfully, but the service request did not pass the audit. Therefore, the service did not move to the **Deployed** state. The **Failed Audit** state is initiated from the **Pending** state. After a service request is deployed successfully, it cannot re-enter the **Failed Audit** state (except if the service request is redeployed). |
| **Failed Deploy** | The cause for a **Failed Deploy** status is that DCS reports that either the upload of the initial configuration file from the routers failed or the download of the configuration update to the routers failed (due to lost connection, faulty password, and so on). |
| **Functional**<br>(valid only for MPLS services) | An MPLS service request moves to **Functional** when the auditor finds the VPN routing and forwarding tables (VRF) for this service and they match with the service intent. This state requires that both the configuration file audit and the routing audit are successful. |
| **Invalid** | **Invalid** indicates that the service request information is incorrect in some way. A service request moves to **Invalid** if the request was either internally inconsistent or not consistent with the rest of the existing network/router configurations (for example, no more interfaces were available on the router). The Provisioning Driver cannot generate configuration updates to service this request. |
| **Lost** | A service request moves to **Lost** when the Auditor cannot find a configuration-level verification of intent in the router configuration files. The service request was in the **Deployed** state, but now some or all router configuration information is missing. A service request can move to the **Lost** state *only* when the service request had been **Deployed**. |

*Table 11-1 Summary of Cisco IP Solution Center Service Request States (continued)*

| Service Request Type | Description |
|---|---|
| **Pending** | A service request moves to **Pending** when the Provisioning Driver determines that the request looks consistent and was able to generate the required configuration updates for this request. **Pending** indicates that the service request has generated the configuration updates and the configuration updates are successfully downloaded to the routers.<br>The Auditor regards pending service requests as new requests and begins the audit. If the service has been freshly provisioned and not yet audited, it is not an error (pending audit). However, if an audit is performed and the service is still pending, it is in an error state. |
| **Requested** | If the service is newly entered and not yet deployed, it is not an error. However, if a Deploy is done and it remains **Requested**, the service is in an error state. |
| **Wait Deploy** | This service request state pertains only when downloading configlets to a server running Cisco Configuration Engine. **Wait Deploy** indicates that the configlet has been generated, but it has not been downloaded to the Cisco Configuration Engine server because the device is not currently online. The configlet is staged in the repository until such time as the Cisco Configuration Engine server notifies ISC that it is up. Configlets in the **Wait Deploy** state are then downloaded to the Cisco Configuration Engine server. |

Table 11-2, "User Operations on ISC Service Requests," describes user operations and their impact on ISC service requests.

*Table 11-2 User Operations on ISC Service Requests*

| User Operations | Description |
|---|---|
| **Decommission** | This user operation removes the service from all devices in the service request. |
| **Force Deploy** | This user operation allows you to **Deploy** a service request from any state except **Closed**. This is equivalent to restarting the state diagram. The service request can move from its current state to any other possible state. However, it does not move to the **Requested** state. |
| **Force Purge** | This user operation removes a service request from the database irrespective of its state. If you **Force Purge** a service request from the ISC repository before first decommissioning the service request, the service remains running on the network (specifically, the configuration remains on the devices on which the service was provisioned), but all record of the service request that created the service is removed from ISC. |
| **Purged** | When a service request is **Purged,** it is removed from the ISC database. |

## Viewing Service Request Details

The service request details include the link endpoints for the service request, the history, and the configlet generated during the service request deployment operation. Use the service request details to help you troubleshoot a problem or error with the service request or to check the commands in the configlet.

From the Service Request Details page, you can view more information about:

- Links—the link endpoint details
- History—Service request history report
- Audit—Audit reports for the link IDs
- Configlets—View the ISC generated configlet for the L2VPN or VPLS service request

The following sections describe the links, history, and configlet details for an L2VPN or VPLS service request. The audit details are described in Auditing Service Requests, page 11-12.

To view service request details, perform the following steps.

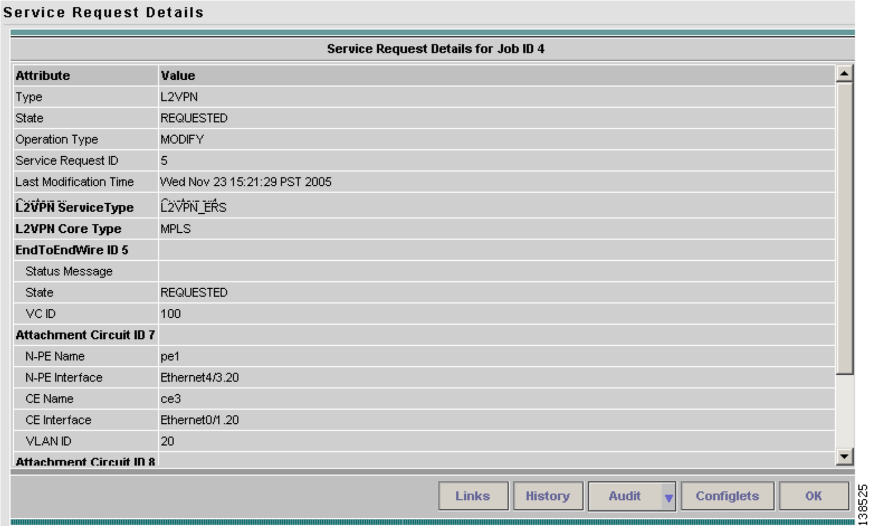
**Step 1** Choose **Service Inventory > Inventory and Connection Manager > Service Requests**.

The Service Requests window appears.

**Step 2** Choose the service request and click **Details**.

The Service Request Details window appears. (See Figure 11-4.)

***Figure 11-4 Example Service Request Details Window***



Service Request Details

Service Request Details for Job ID 4

| Attribute | Value |
|---|---|
| Type | L2VPN |
| State | REQUESTED |
| Operation Type | MODIFY |
| Service Request ID | 5 |
| Last Modification Time | Wed Nov 23 15:21:29 PST 2005 |
| **L2VPN ServiceType** | L2VPN_ERS |
| **L2VPN Core Type** | MPLS |
| **EndToEndWire ID 5** | |
| Status Message | |
| State | REQUESTED |
| VC ID | 100 |
| **Attachment Circuit ID 7** | |
| N-PE Name | pe1 |
| N-PE Interface | Ethernet4/3.20 |
| CE Name | ce3 |
| CE Interface | Ethernet0/1.20 |
| VLAN ID | 20 |
| **Attachment Circuit ID 8** | |

The service request attribute details include the type, transition state, operation type, ID, modification history, customer, and policy name.

## Links

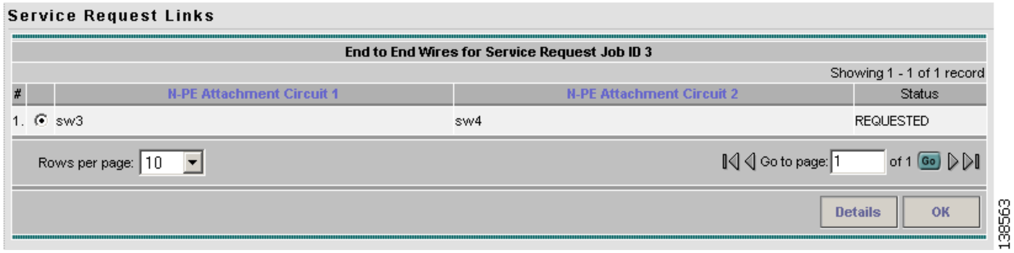
The service request link details include the link endpoints, PE secured interface, VLAN ID, and whether a CE is present.

To see this information, perform the following steps.

**Step 1** Click **Links** on the Service Request Details window. (See Figure 11-4.)

The Service Request Links window appears. (See Figure 11-5.)

*Figure 11-5 Service Request Links*



Service Request Links

End to End Wires for Service Request Job ID 3

Showing 1 - 1 of 1 record

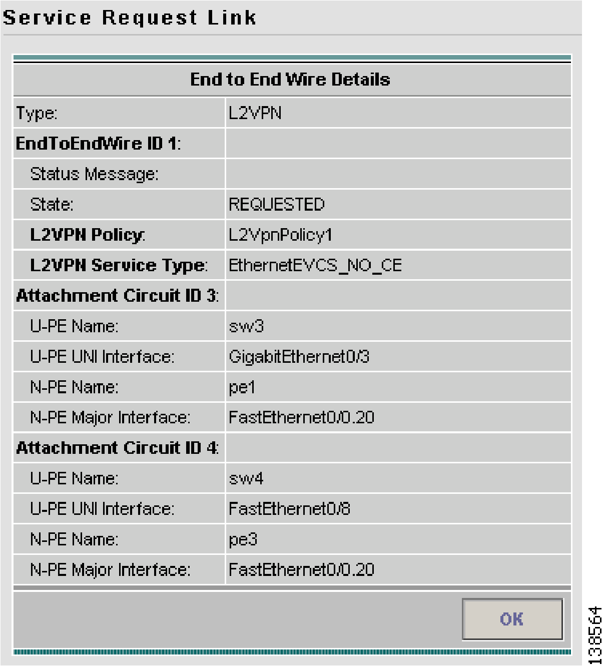
| # | | N-PE Attachment Circuit 1 | N-PE Attachment Circuit 2 | Status |
|---|---|---|---|---|
| 1. | ● | sw3 | sw4 | REQUESTED |

Rows per page: 10

**Step 2** Choose a link and click **Details**.

The Link Details window appears. (See Figure 11-6.)

*Figure 11-6 Link Details Window*



Service Request Link

| End to End Wire Details | |
|---|---|
| Type: | L2VPN |
| **EndToEndWire ID 1**: | |
| Status Message: | |
| State: | REQUESTED |
| **L2VPN Policy**: | L2VpnPolicy1 |
| **L2VPN Service Type**: | EthernetEVCS_NO_CE |
| **Attachment Circuit ID 3**: | |
| U-PE Name: | sw3 |
| U-PE UNI Interface: | GigabitEthernet0/3 |
| N-PE Name: | pe1 |
| N-PE Major Interface: | FastEthernet0/0.20 |
| **Attachment Circuit ID 4**: | |
| U-PE Name: | sw4 |
| U-PE UNI Interface: | FastEthernet0/8 |
| N-PE Name: | pe3 |
| N-PE Major Interface: | FastEthernet0/0.20 |

**Step 3** Click **OK** to return to the Service Request Links window.

**Step 4** Choose another link to view or click **OK** to return to the Service Request Details window.

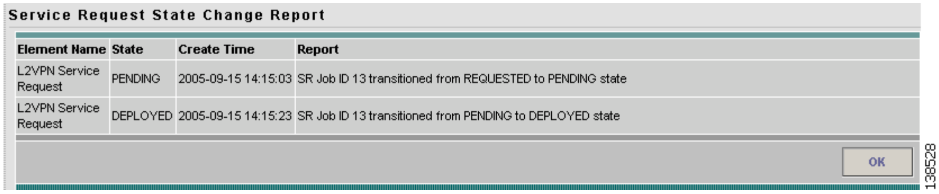
## History

To view history information about the service request, perform the following steps.

**Step 1** Click **History** on the Service Request Details window. (See Figure 11-4.)

The Service Request State Change Report window appears. (See Figure 11-7.)

***Figure 11-7 Service Request State Change Report***



Service Request State Change Report

| Element Name | State | Create Time | Report |
|---|---|---|---|
| L2VPN Service Request | PENDING | 2005-09-15 14:15:03 | SR Job ID 13 transitioned from REQUESTED to PENDING state |
| L2VPN Service Request | DEPLOYED | 2005-09-15 14:15:23 | SR Job ID 13 transitioned from PENDING to DEPLOYED state |

The history reports lists the following information about the service request:

- Element Name—the device, interface, and subinterfaces participating in this service request
- State—the transition states the element has gone through
- Create Time—the time the element was created for this service request
- Report—the action taken by ISC for the element in this service request

**Step 2** Click **OK** to return to the Service Request Details window.

## Configlets

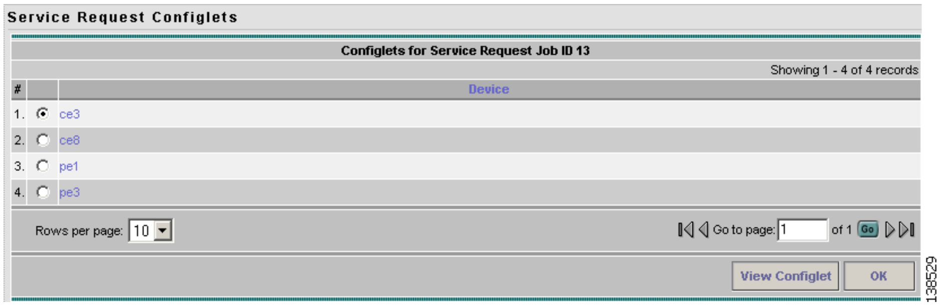
After you deploy the service request, ISC generates Cisco IOS commands to turn on L2VPN or VPLS Services on all the network devices that participate in the service request.

To view the configlets that are generated, perform the following steps.

**Step 1** Click **Configlets** on the Service Request Details window. (See Figure 11-4.)

You see a list of network devices for which a configlet was generated. (See Figure 11-8.)

***Figure 11-8 Service Request Configlets***



Service Request Configlets

Configlets for Service Request Job ID 13

Showing 1 - 4 of 4 records

| # | | Device |
|---|---|---|
| 1. | ● | ce3 |
| 2. | ○ | ce8 |
| 3. | ○ | pe1 |
| 4. | ○ | pe3 |

Rows per page: 10

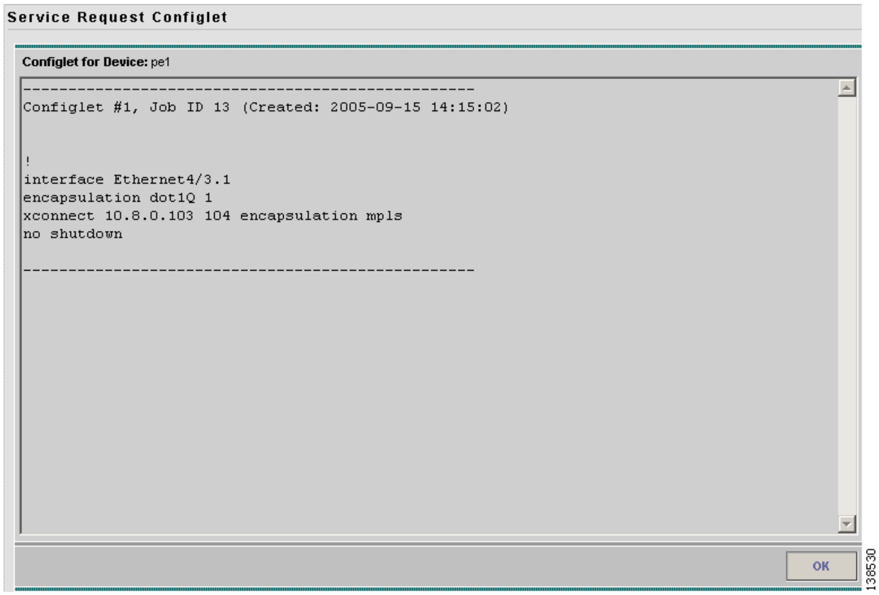
Go to page: 1 of 1

**Step 2** Choose the device for which you want to view the configlet.

**Step 3** Click **View Configlet**.

The Configlet for Device window appears. (See Figure 11-9.)

*Figure 11-9 L2VPN or VPLS Configlet Example*



```
--------------------------------------------------
Configlet #1, Job ID 13 (Created: 2005-09-15 14:15:02)


!
interface Ethernet4/3.1
encapsulation dot1Q 1
xconnect 10.8.0.103 104 encapsulation mpls
no shutdown

--------------------------------------------------
```

The device configlet shows all commands downloaded to the device configuration during the service request deployment operation.

**Step 4** Click **OK** to exit.

# Monitoring Service Requests

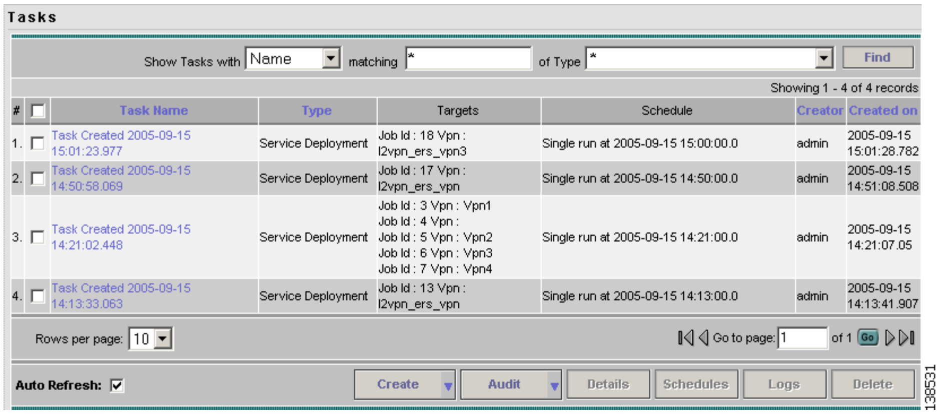
To monitor a service request that is being deployed, you must use the task logs to help you troubleshoot why a service request has failed or to find more details about a service request.

To monitor a service request, perform the following steps.

**Step 1** Choose **Monitoring > Task Manager**.

The Tasks window appears. (See Figure 11-10.)

*Figure 11-10 Tasks Window*



Tasks

Show Tasks with Name matching * of Type *

Showing 1 - 4 of 4 records

| # | □ | Task Name | Type | Targets | Schedule | Creator | Created on |
|---|---|---|---|---|---|---|---|
| 1. | □ | Task Created 2005-09-15 15:01:23.977 | Service Deployment | Job Id : 18 Vpn : l2vpn_ers_vpn3 | Single run at 2005-09-15 15:00:00.0 | admin | 2005-09-15 15:01:28.782 |
| 2. | □ | Task Created 2005-09-15 14:50:58.069 | Service Deployment | Job Id : 17 Vpn : l2vpn_ers_vpn | Single run at 2005-09-15 14:50:00.0 | admin | 2005-09-15 14:51:08.508 |
| 3. | □ | Task Created 2005-09-15 14:21:02.448 | Service Deployment | Job Id : 3 Vpn : Vpn1<br>Job Id : 4 Vpn :<br>Job Id : 5 Vpn : Vpn2<br>Job Id : 6 Vpn : Vpn3<br>Job Id : 7 Vpn : Vpn4 | Single run at 2005-09-15 14:21:00.0 | admin | 2005-09-15 14:21:07.05 |
| 4. | □ | Task Created 2005-09-15 14:13:33.063 | Service Deployment | Job Id : 13 Vpn : l2vpn_ers_vpn | Single run at 2005-09-15 14:13:00.0 | admin | 2005-09-15 14:13:41.907 |

Rows per page: 10 Go to page: 1 of 1

Auto Refresh: ☑

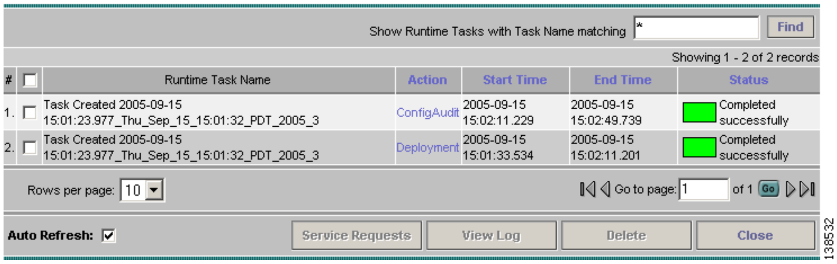
**Step 2** Click **Find** to refresh the window.

The task that is executing will be the first in the list of tasks that being performed in ISC.

**Step 3** Choose the task you want to monitor and click **Logs**.

The Task Logs window appears. (See Figure 11-11.)

*Figure 11-11 Task Logs*



Show Runtime Tasks with Task Name matching *

Showing 1 - 2 of 2 records

| # | □ | Runtime Task Name | Action | Start Time | End Time | Status |
|---|---|---|---|---|---|---|
| 1. | □ | Task Created 2005-09-15 15:01:23.977_Thu_Sep_15_15:01:32_PDT_2005_3 | ConfigAudit | 2005-09-15 15:02:11.229 | 2005-09-15 15:02:49.739 | 🟩 Completed successfully |
| 2. | □ | Task Created 2005-09-15 15:01:23.977_Thu_Sep_15_15:01:32_PDT_2005_3 | Deployment | 2005-09-15 15:01:33.534 | 2005-09-15 15:02:11.201 | 🟩 Completed successfully |

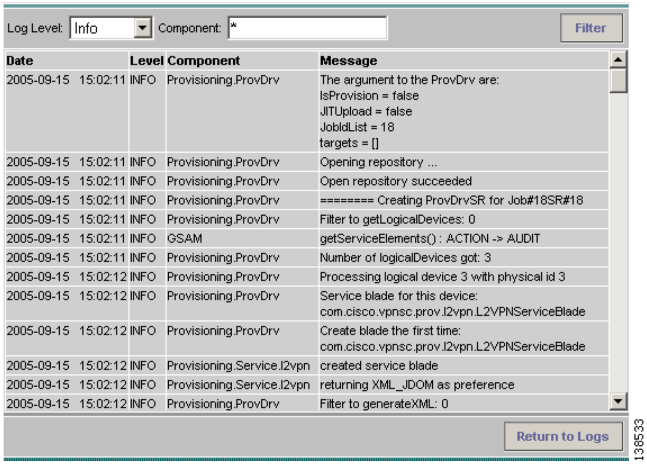
Rows per page: 10 Go to page: 1 of 1

Auto Refresh: ☑

**Step 4** Choose the run-time task that you want to monitor and click **View Log**.

A window like the one shown in Figure 11-12 appears.

*Figure 11-12 Task Logs*



Log Level: Info Component: *

| Date | Level | Component | Message |
|---|---|---|---|
| 2005-09-15 15:02:11 | INFO | Provisioning.ProvDrv | The argument to the ProvDrv are:<br>IsProvision = false<br>JITUpload = false<br>JobIdList = 18<br>targets = [] |
| 2005-09-15 15:02:11 | INFO | Provisioning.ProvDrv | Opening repository ... |
| 2005-09-15 15:02:11 | INFO | Provisioning.ProvDrv | Open repository succeeded |
| 2005-09-15 15:02:11 | INFO | Provisioning.ProvDrv | ======== Creating ProvDrvSR for Job#18SR#18 |
| 2005-09-15 15:02:11 | INFO | Provisioning.ProvDrv | Filter to getLogicalDevices: 0 |
| 2005-09-15 15:02:11 | INFO | GSAM | getServiceElements() : ACTION -> AUDIT |
| 2005-09-15 15:02:11 | INFO | Provisioning.ProvDrv | Number of logicalDevices got: 3 |
| 2005-09-15 15:02:12 | INFO | Provisioning.ProvDrv | Processing logical device 3 with physical id 3 |
| 2005-09-15 15:02:12 | INFO | Provisioning.ProvDrv | Service blade for this device:<br>com.cisco.vpnsc.prov.l2vpn.L2VPNServiceBlade |
| 2005-09-15 15:02:12 | INFO | Provisioning.ProvDrv | Create blade the first time:<br>com.cisco.vpnsc.prov.l2vpn.L2VPNServiceBlade |
| 2005-09-15 15:02:12 | INFO | Provisioning.Service.l2vpn | created service blade |
| 2005-09-15 15:02:12 | INFO | Provisioning.Service.l2vpn | returning XML_JDOM as preference |
| 2005-09-15 15:02:12 | INFO | Provisioning.ProvDrv | Filter to generateXML: 0 |

**Step 5** Choose the log level from the drop-down list and click **Filter**.

The log levels are All, Severe, Warning, Info, Config, Fine, Finer, and Finest.

**Step 6** Click **Return to Logs**.

**Step 7** Click **Close** in the Task Logs window.

# Auditing Service Requests

Each time a service request is deployed in the Cisco IP Solution Center (ISC), a configuration audit occurs. You can view the results of these in configuration audit reports. Use configuration audits and reports to verify that the network devices have the correct configuration for the services provided.

A configuration audit occurs automatically each time you deploy a service request. During this configuration audit, ISC verifies that all Cisco IOS commands are present and that they have the correct syntax. An audit also verifies that there were no errors during deployment.

The configuration audit verifies the service request deployment by examining the commands configured by the service request on the target devices. If the device configuration does not match what is defined in the service request, the audit flags a warning and sets the service request to a **Failed Audit** or **Lost** state.

You can create audit reports for new or existing service requests.

- Audit new services—This type of audit is for service requests that have just been deployed. The audit identifies problems with the configuration files downloaded to the devices.
- Audit existing services—This type of audit checks and evaluates the configuration of deployed service requests to see if the service request is still in effect.

We recommend that you schedule a service request audit on a regular basis to verify the state of the network provisioning requests.

This section describes how to manually generate a configuration audit and view the audit report.

To view a configuration audit report, perform the following steps.

**Step 1** Choose **Service Inventory > Inventory and Connection Manager > Service Requests**.

The Service Requests window appears.

**Step 2** Choose an service request for the configuration audit.

**Step 3** Click **Details**.

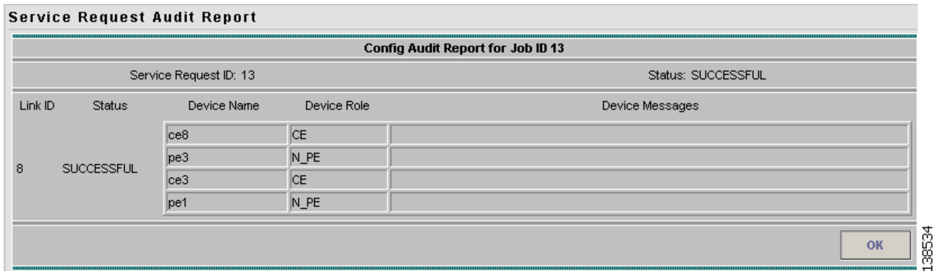
The Service Request Details window appears.

**Step 4** Click **Audit**.

**Step 5** Click **Config.**

The Service Request Audit window appears. Figure 11-13 shows an example of a successful configuration audit.

*Figure 11-13 Service Request Audit Report—Successful*



Service Request Audit Report

Config Audit Report for Job ID 13

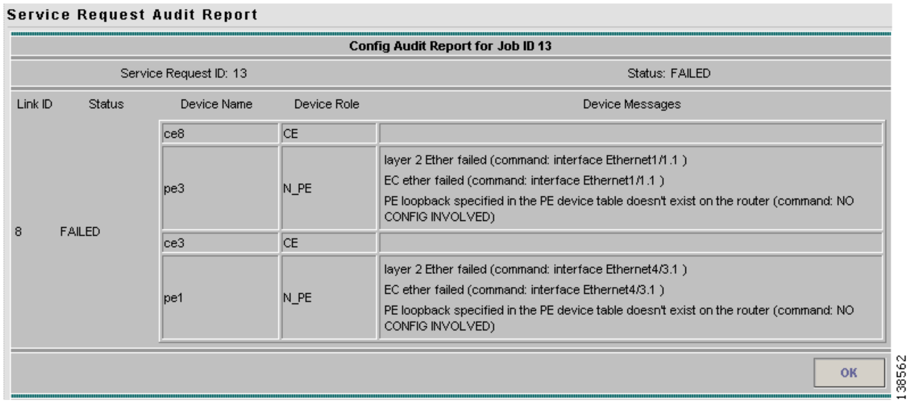
Service Request ID: 13 Status: SUCCESSFUL

| Link ID | Status | Device Name | Device Role | Device Messages |
|---|---|---|---|---|
| 8 | SUCCESSFUL | ce8 | CE | |
| | | pe3 | N_PE | |
| | | ce3 | CE | |
| | | pe1 | N_PE | |

This window lists the device name and role, and a message regarding the status of your configuration audit.

If the audit is unsuccessful, the message field lists details on the failed audit. Figure 11-14 shows an example of a failed audit message for an service request.

*Figure 11-14 Service Request Audit Report—Failed*



Service Request Audit Report

Config Audit Report for Job ID 13

Service Request ID: 13 Status: FAILED

| Link ID | Status | Device Name | Device Role | Device Messages |
|---|---|---|---|---|
| 8 | FAILED | ce8 | CE | |
| | | pe3 | N_PE | layer 2 Ether failed (command: interface Ethernet1/1.1 )<br>EC ether failed (command: interface Ethernet1/1.1 )<br>PE loopback specified in the PE device table doesn't exist on the router (command: NO CONFIG INVOLVED) |
| | | ce3 | CE | |
| | | pe1 | N_PE | layer 2 Ether failed (command: interface Ethernet4/3.1 )<br>EC ether failed (command: interface Ethernet4/3.1 )<br>PE loopback specified in the PE device table doesn't exist on the router (command: NO CONFIG INVOLVED) |

The audit failure message indicates missing commands and configuration issues. Carefully review the information in the message field. If the audit fails, you must correct all errors and redeploy the service request.

**Step 6** Click **OK** to return to the Service Request Details window.

CHAPTER 12

# Using Autodiscovery for L2 Services

All discovery steps are integrated in a discovery workflow, controlled from the ISC GUI. This is accessed in ISC through **Service Inventory > Discovery**. The following discovery features are supported:

- File-based device discovery is supported.
- Rules-based device role assignment is supported.
- Discovery progress messages and logs are viewable in the GUI to keep track of various discovery stages.
- Bulk creation of Provider, Customer, Site, and Region objects is available through an XML data file.

For detailed steps on using the autodiscovery feature in ISC, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*.

CHAPTER 13

# Generating L2 and VPLS Reports

This chapter provides information on generating L2 and VPLS reports. It contains the following sections:

## Overview

The ISC reporting GUI is used across multiple ISC modules, including L2 and VPLS. For a general coverage of using the reports GUI, running reports, using the output from reports, and creating customized reports, see "Monitoring" chapter in the *Cisco IP Solution Center Infrastructure Reference, 6.0*. The rest of this chapter provides information about the L2 and VPLS reports available in ISC.

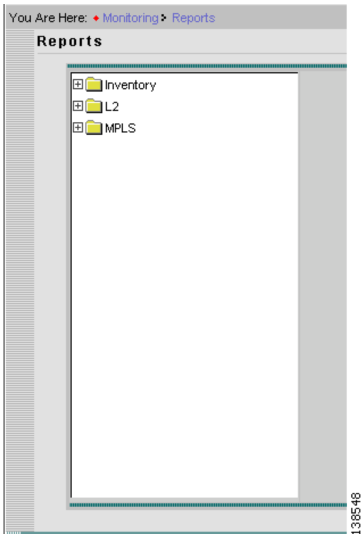
## Accessing L2 and VPLS Reports

Perform the following steps to access the L2 and VPLS reports.

**Step 1** To access the reports framework in the ISC GUI, choose **Monitoring > Reports**.

The Reports window appears. (See Figure 13-1.)

*Figure 13-1 Reports Window*



You Are Here: ◆ Monitoring ▸ Reports

Reports

**Step 2** Click the L2 folder to display the available L2 and VPLS reports.

**Step 3** Click the icon of a report to bring up the window associated with that report.

Details on each of the reports are provided in L2 and VPLS Reports, page 13-2.

# L2 and VPLS Reports

This section provides details on the following L2 and VPLS reports:

- L2 End-to-End Wire Report, page 13-3
- L2 PE Service Report, page 13-6
- L2 VPN Report, page 13-7
- VPLS Attachment Circuit Report, page 13-8
- VPLS PE Service Report, page 13-10
- VPLS VPN Report, page 13-11

**Note** Several sample reports are provided in the L2 reports folder. These reports begin with the title **SAMPLE-**. These reports are provided for informational purposes only. They are untested and unsupported. You might want to use them as a basis for creating your own custom reports. For more information, see Creating Custom L2 and VPLS Reports, page 13-11.

The following information is provided for each report:

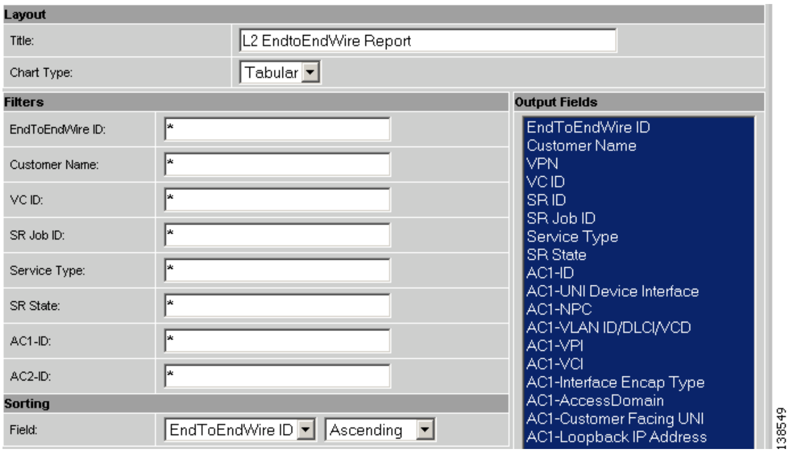
- Description or purpose of the report.
- An illustration of the report window.
- List of filter values and descriptions.
- List of output values and descriptions.

# L2 End-to-End Wire Report

An L2 end-to-end wire is a point-to-point connection containing two attachment circuits. The L2 EndtoEndWire report displays the services that are running on L2 end-to-end connections. You can use this report to view all the services and respective attachment circuit attributes for each connection.

Click the L2 EndtoEndWire Report icon to bring up the window for this report. (See Figure 13-2.)

***Figure 13-2 L2 EndtoEndWire Report***



**Layout**

| | |
|---|---|
| Title: | L2 EndtoEndWire Report |
| Chart Type: | Tabular |

**Filters**

| | |
|---|---|
| EndToEndWire ID: | * |
| Customer Name: | * |
| VC ID: | * |
| SR Job ID: | * |
| Service Type: | * |
| SR State: | * |
| AC1-ID: | * |
| AC2-ID: | * |

**Sorting**

| | | |
|---|---|---|
| Field: | EndToEndWire ID | Ascending |

**Output Fields**

EndToEndWire ID
Customer Name
VPN
VC ID
SR ID
SR Job ID
Service Type
SR State
AC1-ID
AC1-UNI Device Interface
AC1-NPC
AC1-VLAN ID/DLCI/VCD
AC1-VPI
AC1-VCI
AC1-Interface Encap Type
AC1-AccessDomain
AC1-Customer Facing UNI
AC1-Loopback IP Address

Filter Values:

- **EndToEndWire ID**—End-to-end wire identification number.
- **Customer Name**—Name of the customer.
- **VC ID**—Virtual circuit identification number.
- **SR Job ID**—Service request job identification number.
- **Service Type**—Type of service. Values can be:
  - ATM
  - ATM_NO_CE
  - FRAME_RELAY
  - FRAME_RELAY_NO_CE
  - L2VPN_ERS
  - L2VPN_ERS_NO_CE
  - L2VPN_EWS
  - L2VPN_EWS_NO_CE
- **SR State**—Service request state. Values can be:
  - BROKEN
  - DEPLOYED
  - FAILED_AUDIT
  - FAILED_DEPLOY
  - FUNCTIONAL

  - INVALID
  - LOST
  - PENDING
  - REQUESTED
  - WAIT_DEPLOY
- **AC1-ID**—First attachment circuit (AC1) identification number.
- **AC2-ID**—Second attachment circuit (AC2) identification number.

Output Values:

- **EndToEndWire ID**—End-to-end wire identification number.
- **Customer Name**—Name of the customer.
- **VPN**—Name of the VPN.
- **VC ID**—Virtual circuit identification number.
- **SR ID**—Service request identification number.
- **SR Job ID**—Service request job identification number.
- **Service Type**—Type of service.
- **SR State**—Service request state.

**Note** The **SR State** output does not list service requests in the **CLOSED** state. Service requests in other states are listed, as determined by the filter values.

- **AC1-ID**—Identification number of the first attachment circuit (AC1).
- **AC1-UNI Device Interface**—UNI device interface of the first attachment circuit (AC1).
- **AC1-NPC**—Named physical circuit for the first attachment circuit (AC1).
- **AC2-VLAN ID/DLCI/VCD**—VLAN identification number, DLCI (data-link connection identifier) or VCD (virtual circuit descriptor) of the first attachment circuit (AC1).
- **AC1-VPI**—Virtual path identifier for the first attachment circuit (AC1).
- **AC1-VCI**—Virtual channel identifier for the first attachment circuit (AC1).
- **AC1-Interface Encap Type**—Encapsulation type used for the first attachment circuit (AC1).
- **AC1-AccessDomain**—Access domain name for the first attachment circuit (AC1).
- **AC1-Customer Facing UNI**—Customer-facing UNI port of the first attachment circuit (AC1).
- **AC1-Loopback IP Address**—Loop back address for the first attachment circuit (AC1).
- **AC1-STP Shutdown Threshold**—Spanning Tree Protocol shutdown threshold (in packets/second) for the first attachment circuit (AC1).
- **AC1-VTP Shutdown Threshold**—VLAN Trunk Protocol shutdown threshold (in packets/second) for the first attachment circuit (AC1).
- **AC1-CDP Shutdown Threshold**—Cisco Discovery Protocol shutdown threshold (in packets/second) for the first attachment circuit (AC1).
- **AC1-STP Drop Threshold**—Spanning Tree Protocol drop threshold (in packets/second) for the first attachment circuit (AC1).

- **AC1-CDP Drop Threshold**—Cisco Discovery Protocol drop threshold (in packets/second) for the first attachment circuit (AC1).
- **AC1-VTP Drop Threshold**—VLAN Trunk Protocol drop threshold (in packets/second) for the first attachment circuit (AC1).
- **AC1-UNI Recovery Interval**—Recovery interval (in seconds) of the UNI port for the first attachment circuit (AC1).
- **AC1-UNI Speed**—UNI port speed for the first attachment circuit (AC1).
- **AC1-UNI Shutdown—**Shutdown status of the UNI port for the first attachment circuit (AC1).
- **AC1-UNI PortSecurity—**Status of UNI port security for the first attachment circuit (AC1).
- **AC1-UNI Duplex**—Duplex status (none, full, half, or auto) of the UNI port for the first attachment circuit (AC1).
- **AC1-Maximum MAC Address**—Maximum MAC addresses allowed on the UNI port for the first attachment circuit (AC1).
- **AC1-UNI Aging—**Length of time, in seconds, that MAC addresses can stay in the UNI port security table for the first attachment circuit (AC1).
- **AC2-ID**—Second attachment circuit (AC2) identification number.
- **AC2-UNI Device Interface**—UNI device interface of the second attachment circuit (AC2).
- **AC2-NPC**—Named physical circuit for the second attachment circuit (AC2).
- **AC2-VLAN ID/DLCI/VCD**—The VLAN ID, DLCI or VCD of the second attachment circuit (AC2).
- **AC2-VPI**—Virtual path identifier for the first attachment circuit (AC2).
- **AC2-VCI**—Virtual channel identifier for the first attachment circuit (AC2).
- **AC2-Interface Encap Type**—Encapsulation type used for the second attachment circuit (AC2).
- **AC2-AccessDomain**—Access domain name for the second attachment circuit (AC2).
- **AC2-Customer Facing UNI—**Customer-facing UNI port of the second attachment circuit (AC2).
- **AC2-Loopback IP Address—**Loop back address for the second attachment circuit (AC2).
- **AC2-STP Shutdown Threshold**—Spanning Tree Protocol shutdown threshold for the second attachment circuit (AC2).
- **AC2-VTP Shutdown Threshold**—VLAN Trunk Protocol shutdown threshold for the second attachment circuit (AC2).
- **AC2-CDP Shutdown Threshold**—Cisco Discovery Protocol shutdown threshold for the second attachment circuit (AC2).
- **AC2-STP Drop Threshold**—Spanning Tree Protocol drop threshold for the second attachment circuit (AC2).
- **AC2-CDP Drop Threshold**—Cisco Discovery Protocol drop threshold for the second attachment circuit.
- **AC2-VTP Drop Threshold**—VLAN Trunk Protocol drop threshold for the second attachment circuit (AC2).
- **AC2-UNI Recovery Interval**—Recovery interval of the UNI port for the second attachment circuit (AC2).
- **AC2-UNI Speed**—UNI port speed for the second attachment circuit (AC2).
- **AC2-UNI Shutdown**—Shutdown status of the UNI port for the second attachment circuit (AC2).

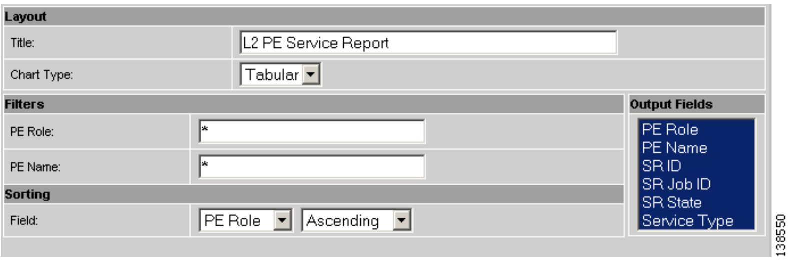
- **AC2-UNI PortSecurity**—Status of UNI port security for the second attachment circuit (AC2).
- **AC2-UNI Duplex**—Duplex status (none, full, half, or auto) of the UNI port for the second attachment circuit (AC2).
- **AC2-Maximum MAC Address**—Maximum MAC addresses allowed on the UNI port for the second attachment circuit (AC2).
- **AC2-UNI Aging**—Length of time, in seconds, that MAC addresses can stay in the UNI port security table for the second attachment circuit (AC2).

# L2 PE Service Report

The L2 PE Service report allows you to choose PEs and display their roles (for example, N-PE, U-PE or PE-AGG) and L2-related services that are running on them.

Click the L2 PE Service Report icon to bring up the window for this report. (See Figure 13-3.)

*Figure 13-3 L2 PE Service Report*



Layout

Title: L2 PE Service Report

Chart Type: Tabular

Filters

PE Role: *

PE Name: *

Sorting

Field: PE Role Ascending

Output Fields

PE Role
PE Name
SR ID
SR Job ID
SR State
Service Type

Filter Values:

- **PE Role**—PE device role (N-PE, U-PE, or PE-AGG).
- **PE Name**—PE device name.

Output Values:

- **PE Role**—PE device role (N-PE, U-PE, or PE-AGG).
- **PE Name**—PE device name.
- **SR ID**—Service request identification number.
- **SR Job ID**—Service request job identification number.
- **SR State**—Service request state.

**Note** The **SR State** output does not list service requests in the **CLOSED** state. Service requests in other states are listed, as determined by the filter values.

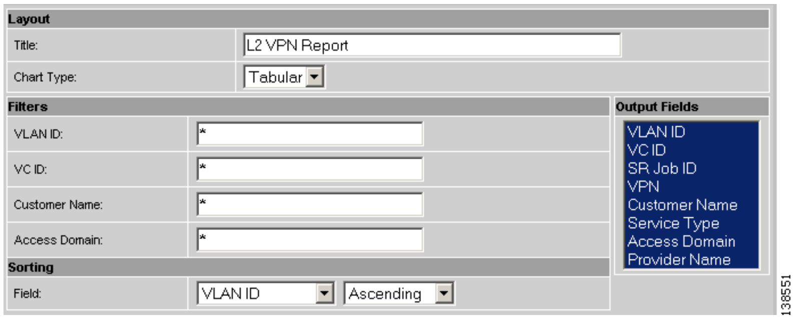
- **Service Type**—Type of service.

# L2 VPN Report

The L2 VPN Report provides a way to track a VLAN ID and/or VC ID back to the VPN and customer without having to iterate through every link and every VPN service. Given a VLAN ID or VC ID, the respective customer and VPN details are displayed in the report.

Click the L2 VPN Report icon to bring up the window for this report. (See Figure 13-4.)

***Figure 13-4 L2 VPN Report***



**Layout**

| | |
|---|---|
| Title: | L2 VPN Report |
| Chart Type: | Tabular |

**Filters**

| | |
|---|---|
| VLAN ID: | * |
| VC ID: | * |
| Customer Name: | * |
| Access Domain: | * |

**Sorting**

| | | |
|---|---|---|
| Field: | VLAN ID | Ascending |

**Output Fields**

VLAN ID
VC ID
SR Job ID
VPN
Customer Name
Service Type
Access Domain
Provider Name

Filter Values:

- **VLAN ID**—VLAN identification number.
- **VC ID**—Virtual circuit identification number.
- **Customer Name**—Name of the customer.
- **Access Domain**—Access domain name.

Output Values:

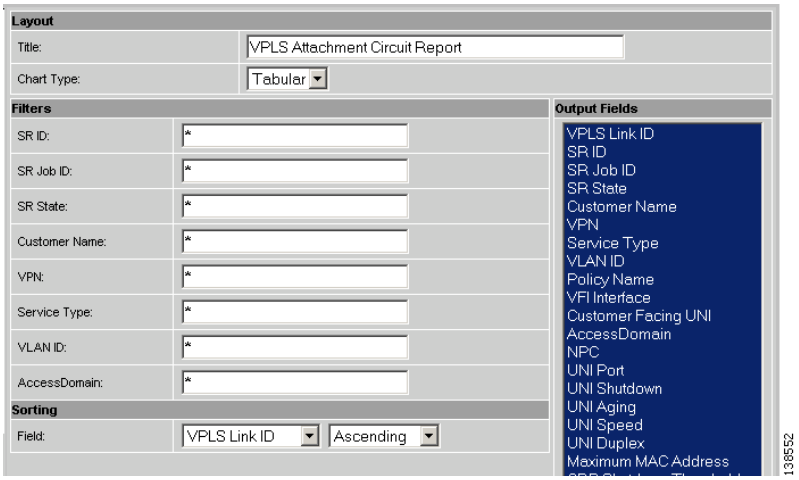
- **VLAN ID**—VLAN identification number.
- **VC ID**—Virtual circuit identification number.
- **SR Job ID**—Service request job identification number
- **VPN**—Name of the VPN.
- **Customer Name**—Name of the customer.
- **Service Type**—Type of service.
- **Access Domain**—Access domain name.
- **Provider Name**—Name of the provider.

# VPLS Attachment Circuit Report

The VPLS Attachment circuit report displays details of attachment circuits for a given customer VPN.

Click the VPLS Attachment Circuit Report icon to bring up the window for this report. (See Figure 13-5.)

*Figure 13-5 VPLS Attachment Circuit Report*



**Layout**

| | |
|---|---|
| Title: | VPLS Attachment Circuit Report |
| Chart Type: | Tabular |

**Filters**

| | |
|---|---|
| SR ID: | * |
| SR Job ID: | * |
| SR State: | * |
| Customer Name: | * |
| VPN: | * |
| Service Type: | * |
| VLAN ID: | * |
| AccessDomain: | * |

**Sorting**

| | | |
|---|---|---|
| Field: | VPLS Link ID | Ascending |

**Output Fields**

VPLS Link ID
SR ID
SR Job ID
SR State
Customer Name
VPN
Service Type
VLAN ID
Policy Name
VFI Interface
Customer Facing UNI
AccessDomain
NPC
UNI Port
UNI Shutdown
UNI Aging
UNI Speed
UNI Duplex
Maximum MAC Address

Filter Values:

- **SR ID**—Service request identification number.
- **SR Job ID**—Service request job identification number.
- **SR State**—Service request state. Values can be:
  - BROKEN
  - DEPLOYED
  - FAILED_AUDIT
  - FAILED_DEPLOY
  - FUNCTIONAL
  - INVALID
  - LOST
  - PENDING
  - REQUESTED
  - WAIT_DEPLOY
- **Customer Name**—Name of the customer.
- **VPN**—Name of the VPN.
- **Service Type**—Type of service. Values can be:
  - VPLS_ERS
  - VPLS_ERS_NO_CE
  - VPLS_EWS

  - VPLS_EWS_NO_CE
- **VLAN ID**—VLAN identification number.
- **AccessDomain**—Access domain name.

Output Values:

- **VPLS Link ID**—VPLS link identification number.
- **SR ID**—Service request identification number
- **SR Job ID**—Service request job identification number.
- **SR State**—Service request state.

**Note** The **SR State** output does not list service requests in the **CLOSED** state. Service requests in other states are listed, as determined by the filter values.

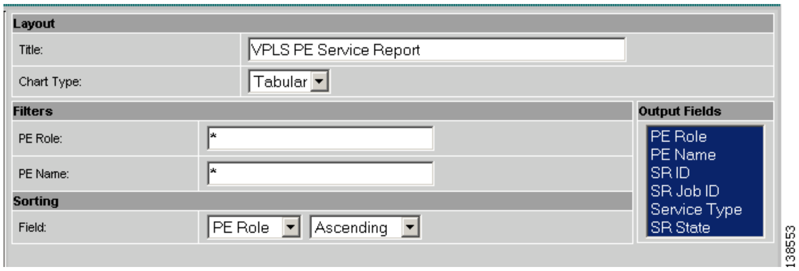
- **Customer Name**—Name of the customer.
- **VPN**—Name of the VPN.
- **Service Type**—Type of service.
- **VLAN ID**—VLAN identification number.
- **Policy Name**—Name of the VPLS policy.
- **VFI Interface**—Virtual forwarding interface name.
- **Customer Facing UNI**—Customer-facing UNI port.
- **AccessDomain**—Access domain name.
- **NPC**—Named physical circuit.
- **UNI Port**—UNI port.
- **UNI Shutdown**—Shutdown status of the UNI port.
- **UNI Aging**—Length of time, in seconds, that MAC addresses can stay in the UNI port security table.
- **UNI Speed**—UNI port speed.
- **UNI Duplex**—Duplex status (none, full, half, or auto) of the UNI port.
- **Maximum MAC Address**—Maximum MAC addresses allowed on the UNI port.
- **CDP Shutdown Threshold**—Cisco Discovery Protocol shutdown threshold (in packets/second) on the UNI port.
- **STP Shutdown Threshold**—Spanning Tree Protocol shutdown threshold (in packets/second) on the UNI port.
- **VTP Shutdown Threshold**—VLAN Trunk Protocol shutdown threshold (in packets/second) on the UNI port.
- **CDP Drop Threshold**—Cisco Discovery Protocol drop threshold (in packets/second) on the UNI port.
- **VTP Drop Threshold**—VLAN Trunk Protocol drop threshold (in packets/second) on the UNI port.
- **STP Drop Threshold**—Spanning Tree Protocol drop threshold (in packets/second) on the UNI port.
- **Recovery Interval**—Recovery interval (in seconds) of the UNI port.

# VPLS PE Service Report

The VPLS PE Service report allows you to choose PEs and display their roles (for example, N-PE, U-PE or PE-AGG) and the VPLS services that are running on them.

Click the VPLS PE Service Report icon to bring up the window for this report. (See Figure 13-6.)

***Figure 13-6 VPLS PE Service Report***



**Layout**

Title: VPLS PE Service Report

Chart Type: Tabular

**Filters**

PE Role: *

PE Name: *

**Sorting**

Field: PE Role Ascending

**Output Fields**

PE Role
PE Name
SR ID
SR Job ID
Service Type
SR State

Filter Values:

- **PE Role**—PE device role (N-PE, U-PE, or PE-AGG).
- **PE Name**—PE device name.

Output Values:

- **PE Role**—PE device role (N-PE, U-PE, or PE-AGG).
- **PE Name**—PE device name.
- **SR ID**—Service request identification number.
- **SR Job ID**—Service request job identification number.
- **Service Type**—Type of service.
- **SR State**—Service request state.

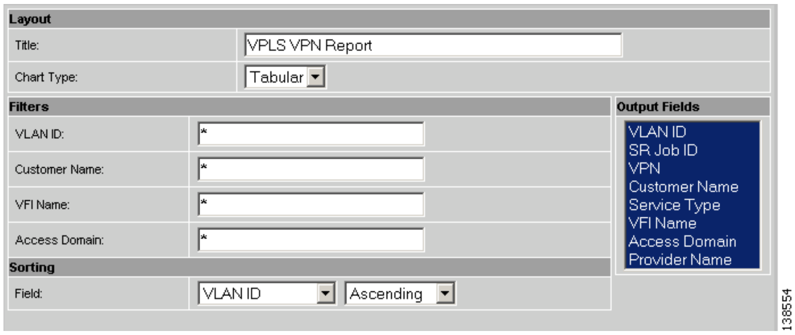
**Note** The **SR State** output does not list service requests in the **CLOSED** state. Service requests in other states are listed, as determined by the filter values.

## VPLS VPN Report

The VPLS VPN report provides a way to track a VLAN ID and/or VFI Name back to the VPN and customer without having to iterate through every link and every VPN service. Given a VLAN ID or VFI name, the respective customer and VPN details are displayed in the report.

Click the VPLS VPN Report icon to bring up the window for this report. (See Figure 13-7.)

*Figure 13-7 VPLS VPN Report*



Layout

Title: VPLS VPN Report

Chart Type: Tabular

Filters

VLAN ID: *

Customer Name: *

VFI Name: *

Access Domain: *

Sorting

Field: VLAN ID Ascending

Output Fields

VLAN ID
SR Job ID
VPN
Customer Name
Service Type
VFI Name
Access Domain
Provider Name

Filter Values:

- **VLAN ID**—VLAN identification number.
- **Customer Name**—Name of the customer.
- **VFI Name**—Virtual forwarding interface name.
- **Access Domain**—Access domain name.

Output Values:

- **VLAN ID**—VLAN identification number.
- **SR Job ID**—Service request job identification number.
- **VPN**—Name of the VPN.
- **Customer Name**—Name of the customer.
- **Service Type**—Type of service.
- **VFI Name**—Virtual forwarding interface name.
- **Access Domain**—Access domain name.
- **Provider Name**—Name of the provider.

# Creating Custom L2 and VPLS Reports

The reports listed in the ISC GUI in the L2 folder are derived from an underlying configuration file. The file is in XML format. You can access the file in the following location:

*$ISC_HOME*/**resources/nbi/reports/ISC/l2_report.xml**

See the "Monitoring" chapter in *Cisco IP Solution Center Infrastructure Reference, 6.0* for details on how to modify report configuration files to create custom reports.

APPENDIX A

# Sample Configlets

This appendix provides sample configlets for L2VPN and Metro Ethernet service provisioning in ISC. It contains the following sections:

# Overview

The configlets provided in this appendix show the CLIs generated by ISC for particular services and features. Each configlet example provides the following information:

- Service
- Feature
- Devices configuration (network role, hardware platform, relationship of the devices and other relevant information)
- Sample configlets for each device in the configuration
- Comments

**Note** The configlets generated by ISC are only the delta between what needs to be provisioned and what currently exists on the device. This means that if a relevant CLI is already on the device, it does not show up in the associated configlet.

**Note** The CLIs shown in bold are the most relevant commands.

**Note** All examples in this appendix assume an MPLS core.

# ERS (EVPL) (Point-to-Point)

## Configuration

- Service: L2VPN/Metro Ethernet.
- Feature: ERS (EVPL) (point-to-point).
- Device configuration:
  - The N-PE is a Cisco 7600 with IOS 12.2(18)SXF, Sup720-3BXL.
    Interface(s): FA8/17.
  - The U-PE is a Cisco 3750ME with 12.2(25)EY1, no port security.
    Interface(s): FA1/0/4 – FA1/0/23.
  - L2VPN point-to-point.

## Configlets

| U-PE | N-PE |
|---|---|
| `vlan 772`<br>`exit`<br>`!`<br>`interface FastEthernet1/0/23`<br>`switchport trunk allowed vlan 500,772`<br>`!`<br>`interface FastEthernet1/0/4`<br>`no cdp enable`<br>`no keepalive`<br>`no ip address`<br>**`switchport trunk allowed vlan 500,772`**<br>**`spanning-tree bpdufilter enable`**<br>**`mac access-group ISC-FastEthernet1/0/4 in`**<br>**`!`**<br>**`mac access-list extended ISC-FastEthernet1/0/4`**<br>**`deny any host 0100.0ccc.cccc`**<br>**`deny any host 0100.0ccc.cccd`**<br>**`deny any host 0100.0ccd.cdd0`**<br>**`deny any host 0180.c200.0000`**<br>**`permit any any`** | `vlan 772`<br>`exit`<br>`!`<br>`interface FastEthernet8/17`<br>`switchport trunk allowed vlan 1,451,653,659,766-768,772,878`<br>`!`<br>**`interface Vlan772`**<br>**`no ip address`**<br>**`description L2VPN ERS`**<br>**`xconnect 99.99.8.99 89027 encapsulation mpls`**<br>**`no shutdown`** |

## Comments

- The N-PE is a 7600 with an OSM or SIP-600 module.
- The U-PE is a generic Metro Ethernet (ME) switch. Customer BPDUs are blocked by the PACL.

# ERS (EVPL) (Point-to-Point, UNI Port Security)

## Configuration

- Service: L2VPN/Metro Ethernet.
- Feature: ERS (EVPL) (point-to-point) with UNI port security.
- Device configuration:
  - The N-PE is a Cisco 7600 with IOS 12.2(18)SXF, OSM.

    Interface(s): FA2/18.
  - The U-PE is a Cisco 3550 with IOS 12.2(25)SEC2. Port security is enabled.

    Interface(s): FA3/31– FA3/23.
  - L2VPN point-to-point.

## Configlets

| U-PE | N-PE |
|---|---|
| `vlan 788`<br>`exit`<br>`!`<br>`interface FastEthernet3/23`<br>`no ip address`<br>`switchport trunk allowed vlan 783,787-788`<br>`!`<br>`interface FastEthernet3/31`<br>`no cdp enable`<br>`no keepalive`<br>`no ip address`<br>`switchport`<br>`switchport trunk encapsulation dot1q`<br>`switchport mode trunk`<br>`switchport trunk allowed vlan none`<br>`switchport trunk allowed vlan 788`<br>**`switchport port-security`**<br>**`switchport nonegotiate`**<br>**`switchport port-security maximum 45`**<br>**`switchport port-security aging time 34`**<br>**`switchport port-security violation shutdown`**<br>**`switchport port-security mac-address 3456.3456.5678`**<br>`spanning-tree bpdufilter enable`<br>`mac access-group ISC-FastEthernet3/31 in`<br>`!`<br>`mac access-list extended ISC-FastEthernet3/31`<br>`deny any host 0100.0ccc.cccc`<br>`deny any host 0100.0ccc.cccd`<br>`deny any host 0100.0ccd.cdd0`<br>`deny any host 0180.c200.0000`<br>**`deny any host 1234.3234.3432`**<br>`permit any any` | `vlan 788`<br>`exit`<br>`!`<br>`interface FastEthernet2/18`<br>`switchport trunk allowed vlan 350,351,430,630,777,780,783,785-788`<br>`!`<br>`interface Vlan788`<br>`no ip address`<br>`description L2VPN ERS with UNI port security`<br>`xconnect 99.99.5.99 89028 encapsulation mpls`<br>`no shutdown` |

## Comments

- The N-PE is a 7600 with an OSM or SIP-600 module.
- The U-PE is a generic Metro Ethernet (ME) switch. The customer BPDUs are blocked by the PACL.

- Various UNI port security commands are provisioned.
- A user-defined PACL entry is added to the default PACL.

# ERS (EVPL) (1:1 VLAN Translation)

## Configuration

- Service: L2VPN/Metro Ethernet.
- Feature: ERS (EVPL) with 1:1 VLAN translation.
- Device configuration:
  - The N-PE is a Cisco 7600 with IOS 12.2(18)SXF, Sup720-3BXL

    Interface(s): FA8/34.
  - The U-PE is a Cisco 3750ME with IOS 12.2(25)EY1. VLAN translation on the NNI port (uplink).

    Interface(s): FA1/0/8 – GI1/1/1.
  - L2VPN point-to-point.

## Configlets

| U-PE | N-PE |
|---|---|
| `!`<br>**`vlan 123`**<br>`exit`<br>`!`<br>`interface FastEthernet1/0/8`<br>`no cdp enable`<br>`no keepalive`<br>`no ip address`<br>**`switchport trunk allowed vlan 123`**<br>`switchport nonegotiate`<br>`switchport port-security maximum 34`<br>`switchport port-security aging time 23`<br>`switchport port-security violation protect`<br>`switchport port-security`<br>`spanning-tree bpdufilter enable`<br>`mac access-group ISC-FastEthernet1/0/8 in`<br>`!`<br>`interface GigabitEthernet1/1/1`<br>`no ip address`<br>`switchport mode trunk`<br>**`switchport trunk allowed vlan 1,123`**<br>**`switchport vlan mapping 123 778`** | `vlan 778`<br>`exit`<br>`!`<br>`interface FastEthernet8/34`<br>`switchport`<br>`switchport trunk encapsulation dot1q`<br>`switchport mode trunk`<br>`switchport trunk allowed vlan 1,778`<br>`!`<br>`interface Vlan778`<br>`no ip address`<br>`description L2VPN ERS 1 to 1 vlan`<br>`translation`<br>`xconnect 99.99.8.99 89032 encapsulation`<br>`mpls`<br>`no shutdown` |

## Comments

- VLAN translation is only for L2VPN (point-to-point) ERS (EVPL).
- In this case, the 1:1 VLAN translation occurs on the U-PE, a 3750. It is provisioned on the NNI (uplink) port.
- The customer VLAN 123 is translated to the provider VLAN 778.

# ERS (EVPL) (2:1 VLAN Translation)

## Configuration

- Service: L2VPN/Metro Ethernet.
- Feature: ERS (EVPL) with VLAN 2:1 translation.Device configuration:
  - The N-PE is a Cisco 7600 with IOS 12.2(18)SXF, Sup720-3BXL

    Interface(s): FA8/34.
  - The U-PE is a Cisco 3750ME with IOS 12.2(25)EY1. VLAN translation on the NNI port (uplink).

    Interface(s): FA1/0/5 – GI1/1/1.
  - L2VPN point-to-point.

## Configlets

| U-PE | N-PE |
|---|---|
| **vlan 567**<br>exit<br>!<br>interface FastEthernet1/0/5<br>no cdp enable<br>no keepalive<br>no ip address<br>switchport<br>**switchport access vlan 567**<br>**switchport mode dot1q-tunnel**<br>switchport trunk allowed vlan none<br>switchport nonegotiate<br>spanning-tree bpdufilter enable<br>mac access-group ISC-FastEthernet1/0/5 in<br>!<br>interface GigabitEthernet1/1/1<br>no ip address<br>switchport trunk allowed vlan 1,123,567<br>**switchport vlan mapping dot1q-tunnel 567 234 779**<br>!<br>mac access-list extended ISC-FastEthernet1/0/5<br>deny any host 0100.0ccc.cccc<br>deny any host 0100.0ccc.cccd<br>deny any host 0100.0ccd.cdd0<br>deny any host 0180.c200.0000<br>permit any any | vlan 779<br>exit<br>!<br>interface FastEthernet8/34<br>switchport trunk allowed vlan 1,778-779<br>!<br>interface Vlan779<br>no ip address<br>description L2VPN ERS 2 to 1 vlan translation<br>xconnect 99.99.8.99 89033 encapsulation mpls<br>no shutdown |

## Comments

- VLAN translation is only for L2VPN (point-to-point) ERS (EVPL).
- In this case, the 2:1 VLAN translation occurs on the U-PE, a 3750. It is provisioned on the NNI (uplink) port.
- The customer VLAN 123 and the provider VLAN 234 (as part of Q -in-Q) are translated to a new provider VLAN 779.

# ERS (Pseudowire Class, E-Line, L2VPN Group Name, IOS XR Device)

## Configuration

- Service: L2VPN/Metro Ethernet.
- Feature: ERS (EVPL).
- Device configuration:
  - The N-PE is a CRS-1 with IOS XR 3.6.1 or later.
  - UNI on N-PE.
  - UNI on U-PE.

## Configlets

**U-PE**

```
!
vlan 700
exit
!
interface FastEthernet1/0/2
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 700
 switchport mode trunk
 switchport nonegotiate
 no keepalive
 mac access-group ISC-FastEthernet1/0/2 in
 no cdp enable
 spanning-tree bpdufilter enable
!
!
interface GigabitEthernet1/0/1
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 700
 switchport mode trunk
 keepalive 10
!
!
mac access-list extended
ISC-FastEthernet1/0/2
deny any host 0100.0ccc.cccc
deny any host 0100.0ccc.cccd
deny any host 0100.0ccd.cdd0
deny any host 0180.c200.0000
permit any any
!
```

**N-PE**

```
!
interface GigabitEthernet0/3/1/1.700
l2transport
 dot1q vlan 700
!
l2vpn
 pw-class PW_AD3-AD7_Customer1
  encapsulation mpls
   transport-mode vlan
   preferred-path interface tunnel-te 1370
fallback disable
  !
 !
 xconnect group L2VPN_Customer1-Gold_class
  p2p GoldPkg_AD3-AD7_Customer1
   interface GigabitEthernet0/3/1/1.700
   neighbor 192.169.105.30 pw-id 1000
    pw-class PW_AD3-AD7_Customer1
   !
  !
```

## Comments

- The N-PE is a CRS-1 with IOS XR 3.7.
- The pseudowire class feature is configured with various associated attributes like encapsulation, transport mode, preferred-path, and fallback option.
- The disable fallback option is required for IOS XR 3.6.1 and optional for IOS XR 3.7 and later.
- The E-Line name (**p2p** command) and L2VPN Group Name (**xconnect group** command) is user configured.

# ERS (EVPL) (NBI Enhancements for L2VPN, IOS Device)

## Configuration

- Service: L2VPN/Metro Ethernet.
- Feature: ERS (EVPL).
- Device configuration:
  - The N-PE is a 12.2(18)SXF with IOS.
  - The U-PE is a 12.2(25)EY4with IOS.
  - UNI on N-PE.
  - UNI on U-PE.

## Configlets

| U-PE | N-PE |
|---|---|
| `!` | `!` |
| `vlan 3200` | `vlan 3300` |
| `exit` | `exit` |
| `!` | `!` |
| `interface FastEthernet1/0/2` | `interface FastEthernet1/0/24` |
| `no cdp enable` | `no cdp enable` |
| `no ip address` | `no ip address` |
| `duplex auto` | `duplex auto` |
| `switchport` | `switchport` |
| `switchport trunk encapsulation dot1q` | `switchport trunk encapsulation dot1q` |
| `switchport mode trunk` | `switchport mode trunk` |
| `switchport trunk allowed vlan none` | `switchport trunk allowed vlan none` |
| `switchport trunk allowed vlan 3200` | `switchport trunk allowed vlan 3300` |
| `switchport nonegotiate` | `switchport nonegotiate` |
| `switchport port-security aging type inactivity` | `switchport port-security aging type inactivity` |
| `switchport port-security maximum 100` | `switchport port-security maximum 100` |
| `switchport port-security aging time 1000` | `switchport port-security aging time 1000` |
| `switchport port-security violation protect` | `switchport port-security violation protect` |
| `switchport port-security` | `switchport port-security` |
| **`storm-control unicast level 1.0`** | **`storm-control unicast level 1.0`** |
| **`storm-control broadcast level 50.0`** | **`storm-control broadcast level 50.0`** |
| **`storm-control multicast level 50.0`** | **`storm-control multicast level 50.0`** |
| `shutdown` | `shutdown` |
| **`keepalive`** | `keepalive` |
| `spanning-tree bpdufilter enable` | `spanning-tree bpdufilter enable` |
| | |
| `!` | `!` |
| `interface GigabitEthernet1/0/1` | **`interface Vlan3300`** |
| `no ip address` | `no ip address` |
| `switchport` | `xconnect 192.169.105.40 7502 encapsulation mpls` |
| `switchport trunk encapsulation dot1q` | `no shutdown` |
| `switchport mode trunk` | `!` |
| `switchport trunk allowed vlan 3200` | |
| `!` | |

## Comments

None.

# ERS (EVPL) or EWS (EPL) (IOS XR Device)

## Configuration

- Service: L2VPN/Metro Ethernet.
- Feature: ERS (EVPL) or EWS (EPL).
- Device configuration(s):
  - The N-PE is a CRS-1 with IOS XR 3.4.2.
  - UNI on N-PE. ERS (EVPL) only.
  - U-PE. EWS (EPL) or ERS (EVPL).

## Configlets

### N-PE

```
<?xml version="1.0" encoding="UTF-8"?>
<Request MajorVersion="1" MinorVersion="0">
  <Set>
    <Configuration Source="CurrentConfig">
      <InterfaceConfigurationTable>
        <InterfaceConfiguration>
          <Naming>
            <Name>GigabitEthernet0/0/0/1.302</Name>
            <Active>act</Active>
          </Naming>
          <InterfaceModeNonPhysical>L2Transport</InterfaceModeNonPhysical>
        </InterfaceConfiguration>
      </InterfaceConfigurationTable>
      <L2VPN>
        <Enabled>true</Enabled>
        <XConnectGroupTable>
          <XConnectGroup>
            <Naming>
              <Name>VPNSC</Name>
            </Naming>
            <Enabled>true</Enabled>
            <P2PXConnectTable>
              <P2PXConnect>
                <Naming>
                  <Name>GigabitEthernet0_0_0_1.302</Name>
                </Naming>
                <Enabled>true</Enabled>
                <AttachmentCircuitTable>
                  <AttachmentCircuit>
                    <Naming>
                      <Name>GigabitEthernet0/0/0/1.302</Name>
                    </Naming>
                    <Enabled>true</Enabled>
                  </AttachmentCircuit>
                </AttachmentCircuitTable>
                <PseudoWireTable>
                  <PseudoWire>
                    <Naming>
                      <Neighbor>
                        <IPV4Address>10.11.13.15</IPV4Address>
                      </Neighbor>
                      <PseudowireID>1005</PseudowireID>
                    </Naming>
                    <PseudoWireParameters/>
                  </PseudoWire>
                </PseudoWireTable>
              </P2PXConnect>
            </P2PXConnectTable>
          </XConnectGroup>
        </XConnectGroupTable>
      </L2VPN>
    </Configuration>
  </Set>
  <Commit/>
</Request>
```

## Comments

- In IOS XR, device configuration is specified in XML format.

- With respect to the XML schemas, different versions of IOS XR generate different XML configlets. However the configurations will be almost identical, except for changes in the XML schema.
- There are different cases to consider. For example, when a service request is decommissioned or modified, the XML configuration will slightly differ.

# ERS (EVPL) and EWS (EPL) (Local Connect on E-Line)

## Configuration

- Service: L2VPN/Metro Ethernet.
- Feature: ERS (EVPL) and EWS (EPL).
- Device configuration:
  - The N-PE is a CRS-1 with IOS XR 3.6 or later.
  - The U-PE is a 12.2(18)SXF with IOS.

## Configlets

| U-PE | N-PE |
|---|---|
|  | interface GigabitEthernet0/0/0/2.559<br>dot1q vlan 559<br>l2transport<br>!<br>interface GigabitEthernet0/0/0/4.559<br>dot1q vlan 559<br>l2transport<br>!<br>l2vpn<br> xconnect group ISC<br>  p2p **cl-test-l2-crs1-1--0--559**<br>   interface GigabitEthernet0/0/0/2.559<br>   interface GigabitEthernet0/0/0/4.559<br>  !<br> !<br>! |

## Comments

- The default E-Line name has changed for local connect configlets.
- The format of the default E-line name is:

  *device_name_with_underscores--VCID--VLANID*

# ERS (EVPL), EWS (EPL), ATM, or Frame Relay (Additional Template Variables for L2VPN, IOS and IOS XR Device)

## Configuration

- Service: L2VPN/Metro Ethernet.
- Feature: ERS (EVPL), EWS (EPL), ATM and Frame Relay.
- Device configuration:
  - The N-PE is a 12.2(18)SXF with IOS for ERS (EVPL), EWS (EPL), Frame Relay service.
  - The N-PE is a CRS-1 with IOS XR 3.6 or later for ERS (EVPL), EWS (EPL) service; and IOS XR 3.7 or later for ATM service (ATM port mode).
  - The U-PE is a 12.2(25)EY4 with IOS for ERS (EVPL) or EWS (EPL) service.

## Configlets

| U-PE | N-PE |
|---|---|
| (None) | Template Content:<br>`interface Loopback0`<br>`description`<br>`LocalLoopbackAddress=$L2VPNLocalLoopback`<br>`LocalHostName=$L2VPNLocalHostName`<br>`RemoteLoopbackAddress=$L2VPNRemoteLoopback`<br>`RemoteHostName=$L2VPNRemoteHostName`<br>Configlets:<br>`interface Loopback0`<br>`description LocalLoopbackAddress=`<br>`192.169.105.40`<br>`LocalHostName=cl-test-l2-7600-2`<br>`RemoteLoopbackAddress=192.169.105.80`<br>`RemoteHostName= cl-test-l2-7600-4` |

## Comments

- These four variables are supported only on the N-PE.
- The values will be empty for all other device roles (U-PE, PE-AGG, and CE).

# EWS (EPL) (Point-to-Point)

### Configuration

- Service: L2VPN/Metro Ethernet.
- Feature: EWS (EPL) (point-to-point).
- Device configuration:
  - The N-PE is a Cisco 7600 with IOS 12.2(18)SXF, Sup720-3BXL.

    Interface(s): FA8/17.
  - The U-PE is a Cisco 3750ME with IOS 12.2(25)EY1. No port security, no tunneling.

    Interface(s): FA1/0/20 – FA1/0/23.
  - L2VPN point-to-point.
  - Q-in-Q UNI.

### Configlets

| U-PE | N-PE |
|---|---|
| **`system mtu 1522`**<br>`!`<br>`vlan 774`<br>`exit`<br>`!`<br>`interface FastEthernet1/0/20`<br>`no cdp enable`<br>`no keepalive`<br>`switchport`<br>**`switchport access vlan 774`**<br>**`switchport mode dot1q-tunnel`**<br>**`switchport nonegotiate`**<br>**`spanning-tree portfast`**<br>**`spanning-tree bpdufilter enable`**<br>`!`<br>`interface FastEthernet1/0/23`<br>`no ip address`<br>`switchport trunk allowed vlan 774,787-788` | `vlan 774`<br>`exit`<br>`!`<br>`interface FastEthernet8/17`<br>`switchport trunk allowed vlan 1,451,653,659,766-768,772,773-774,878`<br>`!`<br>`interface Vlan774`<br>`no ip address`<br>`description L2VPN EWS`<br>`xconnect 99.99.8.99 89029 encapsulation mpls`<br>`no shutdown` |

### Comments

- The N-PE is a 7600 with a OSM or SIP-600 module. Provisioning is the same as the ERS (EVPL) example.
- The U-PE is a generic Metro Ethernet (ME) switch.
- No PACL provisioned by default. BPDU can be tunneled if desired.
- The system MTU needs to set to 1522 to handle the extra 4 bytes of Q-in-Q frames.

# EWS (EPL) (Point-to-Point, UNI Port Security, BPDU Tunneling)

## Configuration

- Service: L2VPN/Metro Ethernet.
- Feature: EWS (EPL) (point-to-point) with Port security, BPDU tunneling.
- Device configuration:
  - The N-PE is a Cisco 7600 with IOS 12.2(18)SXF, Sup720-3BXL.
  - The U-PE is a Cisco 3750ME with IOS 12.2(25)EY1. No port security, with tunneling.
  - L2VPN point-to-point.
  - Q-in-Q UNI.

## Configlets

| U-PE | N-PE |
|---|---|
| system mtu 1522<br>!<br>vlan 775<br>exit<br>!<br>system mtu 1522<br>!<br>vlan 775<br>exit<br>!<br>interface FastEthernet1/0/19<br>no cdp enable<br>no keepalive<br>switchport<br>switchport access vlan 775<br>switchport mode dot1q-tunnel<br>switchport nonegotiate<br>**switchport port-security maximum 34**<br>**switchport port-security aging time 32**<br>**switchport port-security violation shutdown**<br>**switchport port-security**<br>**l2protocol-tunnel cdp**<br>**l2protocol-tunnel stp**<br>**l2protocol-tunnel vtp**<br>**l2protocol-tunnel shutdown-threshold cdp 88**<br>**l2protocol-tunnel shutdown-threshold stp 99**<br>**l2protocol-tunnel shutdown-threshold vtp 56**<br>**l2protocol-tunnel drop-threshold cdp 56**<br>**l2protocol-tunnel drop-threshold stp 64**<br>**l2protocol-tunnel drop-threshold vtp 34**<br>**storm-control unicast level 34.0**<br>**storm-control broadcast level 23.0**<br>**storm-control multicast level 12.0**<br>spanning-tree portfast<br>spanning-tree bpdufilter enable<br>mac access-group ISC-FastEthernet1/0/19 in<br><br>interface FastEthernet1/0/23<br>no ip address<br>switchport trunk allowed vlan<br>774-775,787-788<br><br>!<br>mac access-list extended<br>ISC-FastEthernet1/0/19<br>no permit any any<br>**deny any host 3456.3456.1234**<br>permit any any | vlan 775<br>exit<br>!<br>interface FastEthernet8/17<br>switchport trunk allowed vlan<br>1,451,653,659,766-768,772,773-775,878<br>!<br>interface Vlan775<br>no ip address<br>description L2VPN EWS<br>xconnect 99.99.8.99 89029 encapsulation<br>mpls<br>no shutdown |

## Comments

- The N-PE is a 7600 with an OSM or SIP-600 module. Provisioning is the same as the ERS (EVPL) example.
- The U-PE is a generic Metro Ethernet (ME) switch.
- PACL with one user-defined entry.
- BPDUs (CDP, STP and VTP) are tunneled through the MPLS core.
- Storm control is enabled for unicast, multicast, and broadcast.

# EWS (EPL) (Hybrid)

## Configuration

- Service: L2VPN/Metro Ethernet.
- Feature: EWS (EPL) hybrid. One side is EWS (EPL) UNI; the other side is ERS (EVPL) NNI.
- Device configuration:
  - The N-PE is a Cisco 7600 with 12.2(18)SXF, Sup720-3BXL.

    Interface(s): FA8/17.
  - The U-PE is a Cisco 3750ME with 12.2(25)EY1. No port security, with tunneling.

    Interface(s): FA1/0/20 – FA1/0/23.
  - L2VPN point-to-point.
  - Q-in-Q UNI.

**Note** The first configlet example is the EWS (EPL) side (UNI). The second configlet is the ERS (EVPL) side (NNI).

## Configlets (EWS)

| U-PE | N-PE |
|---|---|
| system mtu 1522<br>!<br>vlan 775<br>exit<br>!<br>system mtu 1522<br>!<br>vlan 775<br>exit<br>!<br>interface FastEthernet1/0/19<br>no cdp enable<br>no keepalive<br>switchport<br>switchport access vlan 775<br>switchport mode dot1q-tunnel<br>switchport nonegotiate<br>**switchport port-security maximum 34**<br>**switchport port-security aging time 32**<br>**switchport port-security violation shutdown**<br>**switchport port-security**<br>**l2protocol-tunnel cdp**<br>**l2protocol-tunnel stp**<br>**l2protocol-tunnel vtp**<br>**l2protocol-tunnel shutdown-threshold cdp 88**<br>**l2protocol-tunnel shutdown-threshold stp 99**<br>**l2protocol-tunnel shutdown-threshold vtp 56**<br>**l2protocol-tunnel drop-threshold cdp 56**<br>**l2protocol-tunnel drop-threshold stp 64**<br>**l2protocol-tunnel drop-threshold vtp 34**<br>**storm-control unicast level 34.0**<br>**storm-control broadcast level 23.0**<br>**storm-control multicast level 12.0**<br>spanning-tree portfast<br>spanning-tree bpdufilter enable<br>mac access-group ISC-FastEthernet1/0/19 in<br><br>interface FastEthernet1/0/23<br>no ip address<br>switchport trunk allowed vlan<br>774-775,787-788<br><br>!<br>mac access-list extended<br>ISC-FastEthernet1/0/19<br>no permit any any<br>**deny any host 3456.3456.1234**<br>permit any any | vlan 775<br>exit<br>!<br>interface FastEthernet8/17<br>switchport trunk allowed vlan<br>1,451,653,659,766-768,772,773-775,878<br>!<br>interface Vlan775<br>no ip address<br>description L2VPN EWS<br>xconnect 99.99.8.99 89029 encapsulation<br>mpls<br>no shutdown |

## Comments

- This is the EWS (EPL) side (UNI).
- N-PE is 7600 with an OSM or a SIP-600 module. Provisioning is the same as the ERS (EVPL).
- The U-PE is a generic Metro Ethernet (ME) switch.
- PACL with one user-defined entry.
- BPDUs (cdp, stp and vtp) are tunneled through the MPLS core.
- Storm control is enabled for unicast, multicast, and broadcast.

## Configlets (ERS)

| U-PE | N-PE |
|---|---|
| `system mtu 1522`<br><br>`vlan 775`<br>`exit`<br><br>`interface FastEthernet1/17`<br>`switchport trunk allowed vlan`<br>`1,451,653,659,766-768,772,773-775,878`<br><br>`interface FastEthernet1/10`<br>`switchport trunk allowed vlan`<br>`1,451,653,659,766-768,772,773-775,878` | `vlan 775`<br>`exit`<br>`!`<br>`interface FastEthernet8/17`<br>`switchport trunk allowed vlan`<br>`1,451,653,659,766-768,772,773-775,878`<br>`!`<br>`interface Vlan775`<br>`no ip address`<br>`description L2VPN EWS`<br>`xconnect 99.99.8.99 89029 encapsulation`<br>`mpls`<br>`no shutdown` |

## Comments

- This is the ERS (EVPL) side (NNI).
- The N-PE is a 7600 with an OSM or a SIP-600 module. Provisioning is the same as the ERS (EVPL).
- The U-PE is really a PE-AGG. It connects to the wholesale customer as an NNI. Both ports are regular NNI ports.

# EWS (EPL) (Pseudowire Class, E-Line, L2VPN Group Name, IOS XR Device)

## Configuration

- Service: L2VPN/Metro Ethernet.
- Feature: EWS (EPL).
- Device configuration:
  - The N-PE is a CRS-1 with IOS XR 3.6.1 or later.
  - UNI on U-PE.

## Configlets

**U-PE**

```
!
system mtu 1522
!
vlan 700
exit
!
interface FastEthernet1/0/2
switchport
switchport access vlan 700
switchport mode dot1q-tunnel
switchport nonegotiate
no keepalive
no cdp enable
spanning-tree portfast
spanning-tree bpdufilter enable
!
interface GigabitEthernet1/0/1
no ip address
switchport
switchport trunk encapsulation dot1q
switchport trunk allowed vlan 700
switchport mode trunk
!
```

**N-PE**

```
!
interface GigabitEthernet0/3/1/1.700
l2transport
 dot1q vlan 700
!
!
l2vpn
 pw-class PW_AD7-AD3_Cutsomer2
  encapsulation mpls
   transport-mode ethernet
   preferred-path interface tunnel-te 2730
  !
 !
 xconnect group ISC
  p2p cl-test-l2-12404-2--1000
   interface GigabitEthernet0/3/1/1.700
   neighbor 192.169.105.30 pw-id 1000
    pw-class PW_AD7-AD3_Cutsomer2
   !
```

## Comments

- The N-PE is a CRS-1 router with IOS XR 3.7.
- The pseudowire class feature is configured with various associated attributes like encapsulation, transport mode, preferred-path, and fallback option
- The disable fallback option is required for IOS XR 3.6.1 and optional for IOS XR 3.7 and later.
- The E-Line name (**p2p** command) and L2VPN Group Name (**xconnect group** command) is an ISC-generated default value, if user input is not provided.

# EWS (EPL) (NBI Enhancements for L2VPN, IOS Device)

## Configuration

- Service: L2VPN/Metro Ethernet.
- Feature: EWS (EPL).
- Device configuration:
  - The N-PE is a 12.2(18)SXF with IOS.
  - The U-PE is a 12.2(25)EY4with IOS.
  - UNI on N-PE.
  - UNI on U-PE.

## Configlets

| U-PE | N-PE |
|---|---|
| `!`<br>`vlan 3201`<br>`exit`<br>`!`<br>`interface FastEthernet1/0/2`<br>`no cdp enable`<br>`no ip address`<br>`duplex auto`<br>`switchport`<br>`switchport access vlan 3201`<br>`switchport mode dot1q-tunnel`<br>`switchport nonegotiate`<br>`switchport port-security aging type inactivity`<br>`switchport port-security maximum 100`<br>`switchport port-security aging time 1000`<br>`switchport port-security violation protect`<br>`switchport port-security`<br>**`storm-control unicast level 1.0`**<br>**`storm-control broadcast level 50.0`**<br>**`storm-control multicast level 50.0`**<br>`shutdown`<br>**`keepalive`**<br>`spanning-tree bpdufilter enable`<br><br>`!`<br>`interface GigabitEthernet1/0/1`<br>`no ip address`<br>`switchport`<br>`switchport trunk encapsulation dot1q`<br>`switchport mode trunk`<br>`switchport trunk allowed vlan 3201`<br>`!` | `!`<br>`vlan 3301`<br>`exit`<br>`!`<br>`interface FastEthernet1/0/24`<br>`no cdp enable`<br>`no ip address`<br>`duplex auto`<br>`switchport`<br>`switchport access vlan 3301`<br>`switchport mode dot1q-tunnel`<br>`switchport nonegotiate`<br>`switchport port-security aging type inactivity`<br>`switchport port-security maximum 100`<br>`switchport port-security aging time 1000`<br>`switchport port-security violation protect`<br>`switchport port-security`<br>**`storm-control unicast level 1.0`**<br>**`storm-control broadcast level 50.0`**<br>**`storm-control multicast level 50.0`**<br>`shutdown`<br>**`keepalive`**<br>`spanning-tree bpdufilter enable`<br><br>`!`<br>**`interface Vlan3301`**<br>`no ip address`<br>`xconnect 192.169.105.40 7502 encapsulation mpls`<br>`no shutdown`<br>`!` |

## Comments

None.

# ATM over MPLS (VC Mode)

## Configuration

- Service: L2VPN.
- Feature: ATM over MPLS (ATMoMPLS, a type of AToM) in VC mode.
- Device configuration:
  - The N-PE is a Cisco 7200 with IOS 12.0(28)S.
  - No CE.
  - No U-PE.
  - L2VPN point-to-point (ATMoMPLS).
  - C7200 (ATM2/0).

## Configlets

| U-PE | N-PE |
|---|---|
| (None) | `interface ATM2/0.34234 point-to-point`<br>`pvc 213/423 l2transport`<br>`encapsulation aal5`<br>`xconnect 99.99.4.99 89025 encapsulation mpls` |

## Comments

- The N-PE is any MPLS-enabled router.
- L2VPN provisioning is on the ATM VC connection.

# ATM over MPLS (VP Mode)

## Configuration

- Service: L2VPN.
- Feature: ATM over MPLS (ATMoMPLS, a type of AToM) in VP mode.
- Device configuration:
  - The N-PE is a Cisco 7200 with IOS 12.0(28)S.

    Interface(s): ATM2/0.
  - No CE.
  - No U-PE.
  - L2VPN point-to-point (ATMoMPLS).

## Configlets

| U-PE | N-PE |
|---|---|
| (None) | **pseudowire-class ISC-pw-tunnel-123**<br>**encapsulation mpls**<br>**preferred-path interface tunnel123**<br>**disable-fallback**<br>**!**<br>**interface ATM2/0**<br>**atm pvp 131 l2transport**<br>**xconnect 99.99.4.99 89024 pw-class ISC-pw-tunnel-123** |

## Comments

- The N-PE is any MPLS-enabled router.
- L2VPN provisioning is on the ATM VP connection.
- The L2VPN pseudowire is mapped to a TE tunnel.

# ATM (Port Mode, Pseudowire Class, E-Line, L2VPN Group Name, IOS XR Device)

## Configuration

- Service: L2VPN/Metro Ethernet.
- Feature: ATM.
- Device configuration:
  - The N-PE is a CRS-1 with IOS XR 3.7 or later for ATM service (port mode only).
  - UNI on N-PE.

## Configlets

<table>
<tr><th>U-PE</th><th>N-PE</th></tr>
<tr><td>(None)</td><td><pre>
interface ATM0/1/0/0
 description UNIDesc_AC1
 l2transport
 !
!
l2vpn
 pw-class PWClass-1
  encapsulation mpls
   preferred-path interface tunnel-te 500
fallback disable
  !
 !
 xconnect group ISC
  p2p ELine_AC1
   interface ATM0/1/0/0
   neighbor 192.169.105.70 pw-id 100
    pw-class PWClass-1
   !
</pre></td></tr>
</table>

## Comments

- The N-PE is a CRS-1 router.
- The pseudowire class feature is optional and not configured.
- The E-Line name (**p2p** command) and L2VPN Group Name (**xconnect group** command) are user configured.
- Only PORT mode is supported in IOS XR.
- This PORT mode will not generate any specific command, such as **pvp** or **pvc**, on IOS XR devices.
- The ATM interface is included under **xconnect**.

# Frame Relay over MPLS

## Configuration

- Service: L2VPN.
- Feature: Frame Relay over MPLS (FRoMPLS, a type of AToM).
- Device configuration:
  - The N-PE is a Cisco 7200 with IOS 12.0(28)S.

    Interface(s): ATM2/0.
  - No CE.
  - No U-PE.
  - L2VPN point-to-point (ATMoMPLS).

## Configlets

| U-PE | N-PE |
|---|---|
| (None) | `interface Serial1/1`<br>`exit`<br>`!`<br>`connect C1_89001 Serial1/1 135 l2transport`<br>`xconnect 99.99.4.99 89001 encapsulation mpls` |

## Comments

- The N-PE is any MPLS-enabled router.
- L2VPN provisioning is on the serial port for the Frame Relay connection.

# Frame Relay (DLCI Mode)

### Configuration

- Service: L2VPN over a L2TPv3 core.
- Feature: FR in DLCI mode.
- Device configuration:
  - The N-PE is a Cisco 7200 with IOS 12.0(28)S.

    Interface(s): ATM2/0.
  - No CE.
  - No U-PE.
  - L2VPN point-to-point (ATMoMPLS).

### Configlets

| U-PE | N-PE |
|---|---|
| (None) | `pseudowire-class ISC-pw-dynamic-default`<br>`encapsulation l2tpv3`<br>`ip local interface Loopback10`<br>`ip dfbit set`<br>`!`<br>`interface Serial3/2`<br>`encapsulation frame-relay`<br>`exit`<br>`!`<br>`connect ISC_1054 Serial3/2 86 l2transport`<br>`xconnect 10.9.1.1 1054 encapsulation l2tpv3`<br>`pw-class ISC-pw-dynamic-default` |

### Comments

- The N-PE is any L2TPv3 enabled router.
- L2VPN provisioning is on the serial port for the Frame Relay connection.

# VPLS (Multipoint, ERMS/EVP-LAN)

## Configuration

- Service: L2VPN/Metro Ethernet.
- Feature: VPLS (multipoint) ERMS (EVP-LAN).
- Device configuration:
  - The N-PE is a Cisco 7600 with IOS 12.2(18)SXF, Sup720-3BX.L

    Interface(s): FA2/18.
  - The U-PE is a Cisco 3750ME with IOS 12.2(25)EY1. No port security, no tunneling.

    Interface(s): FA1/0/21 – FA1/0/23.
  - VPLS Multipoint VPN with VLAN 767.

## Configlets

| U-PE | N-PE |
|---|---|
| `vlan 767`<br>`exit`<br>`!`<br>`interface FastEthernet1/0/21`<br>`no cdp enable`<br>`no keepalive`<br>`no ip address`<br>`switchport`<br>`switchport trunk encapsulation dot1q`<br>`switchport mode trunk`<br>`switchport trunk allowed vlan none`<br>`switchport trunk allowed vlan 767`<br>`switchport nonegotiate`<br>`spanning-tree bpdufilter enable`<br>`mac access-group ISC-FastEthernet1/0/21 in`<br>`!`<br>`interface FastEthernet1/0/23`<br>`no ip address`<br><br>`mac access-list extended`<br>`ISC-FastEthernet1/0/21`<br>`deny any host 0100.0ccc.cccc`<br>`deny any host 0100.0ccc.cccd`<br>`deny any host 0100.0ccd.cdd0`<br>`deny any host 0180.c200.0000`<br>`permit any any` | **`l2 vfi vpls_ers_1-0 manual`**<br>**`vpn id 89017`**<br>**`neighbor 99.99.10.9 encapsulation mpls`**<br>**`neighbor 99.99.5.99 encapsulation mpls`**<br>`!`<br>`vlan 767`<br>`exit`<br>`!`<br>`interface FastEthernet2/18`<br>`switchport trunk allowed vlan`<br>`350,351,430,630,767,780,783,785-791`<br>`!`<br>**`interface Vlan767`**<br>**`no ip address`**<br>**`description VPLS ERS`**<br>**`xconnect vfi vpls_ers_1-0`**<br>**`no shutdown`** |

## Comments

- The N-PE is a 7600 with OSM or SIP-600 module.
- The VFI contains all the N-PEs (neighbors) that this N-PE talks to.
- The U-PE is a generic Metro Ethernet (ME) switch. The customer BPDUs are blocked by the PACL. The VPLS ERMS (EVP-LAN) UNI is the same as the L2VPN (point-to-point) ERS (EVPL) UNI.
- The SVI (interface 767) refers to the global VFI, which contains multiple peering N-PEs.

# VPLS (Multipoint, EMS/EP-LAN), BPDU Tunneling)

## Configuration

- Service: L2VPN/Metro Ethernet.
- Feature: VPLS (multipoint) EMS (EP-LAN) with BPDU tunneling.
- Device configuration:
  - The N-PE is a Cisco 7600 with IOS 12.2(18)SXF, Sup720-3BXL.

    Interface(s): FA2/18.
  - The U-PE is a Cisco 3750ME with IOS 12.2(25)EY1. No port security, no tunneling.

    Interface(s): FA1/0/12 – FA1/0/23.
  - VPLS Multipoint VPN, with VLAN 767.
  - Q-in-Q UNI.

## Configlets

| U-PE | N-PE |
|---|---|
| system mtu 1522<br>!<br>errdisable recovery interval 33<br>!<br>vlan 776<br>exit<br>!<br>interface FastEthernet1/0/12<br>no cdp enable<br>no keepalive<br>switchport<br>**switchport access vlan 776**<br>**switchport mode dot1q-tunnel**<br>**switchport nonegotiate**<br>**l2protocol-tunnel cdp**<br>**l2protocol-tunnel stp**<br>**l2protocol-tunnel vtp**<br>**l2protocol-tunnel shutdown-threshold cdp 88**<br>**l2protocol-tunnel shutdown-threshold stp 64**<br>**l2protocol-tunnel shutdown-threshold vtp 77**<br>**l2protocol-tunnel drop-threshold cdp 34**<br>**l2protocol-tunnel drop-threshold stp 23**<br>**l2protocol-tunnel drop-threshold vtp 45**<br>no shutdown<br>spanning-tree portfast<br>spanning-tree bpdufilter enable | **l2 vfi vpls_ews-89019 manual**<br>**vpn id 89019**<br>**neighbor 99.99.8.99 encapsulation mpls**<br>**!**<br>**vlan 776**<br>**exit**<br>**!**<br>**interface FastEthernet8/17**<br>**switchport trunk allowed vlan**<br>**1,451,653,659,766-768,772-776,878**<br>**!**<br>**interface Vlan776**<br>**no ip address**<br>**description VPLS EWS**<br>**xconnect vfi vpls_ews-89019**<br>**no shutdown** |

## Comments

- The N-PE is a 7600 with an OSM or SIP-600 module.
- The VFI contains all the N-PEs (neighbors) that this N-PE talks to.
- The VPLS EMS (EP-LAN) UNI is the same as L2VPN (point-to-point) EWS (EPL) UNI.
- The SVI is the same as VPLS ERS (EVP-LAN) SVI.

# FlexUNI/EVC (Pseudowire Core Connectivity, UNI Port Security)

## Configuration

- Service: FlexUNI (EVC)/Metro Ethernet.
- Feature: FlexUNI/EVC with pseudowire core connectivity, with UNI port security.
- Device configuration:
  - The N-PE is a Cisco 7600 with IOS 12.2(33)SRB3.
    Interface(s): GI2/0/0.
  - The U-PE is a Cisco 3750ME with IOS 12.2(25)EY2. Port security is enabled.
    Interface(s): FA1/14– FA3/23.

## Configlets

| U-PE | N-PE |
|---|---|
| vlan 788<br>exit<br>!<br>interface FastEthernet3/23<br>no ip address<br>switchport trunk allowed vlan 783,787-788<br>!<br>interface FastEthernet1/14<br>no cdp enable<br>no keepalive<br>no ip address<br>switchport<br>**switchport trunk encapsulation dot1q**<br>**switchport mode trunk**<br>switchport trunk allowed vlan none<br>**switchport trunk allowed vlan 788**<br>**switchport port-security**<br>switchport nonegotiate<br>switchport port-security maximum 45<br>switchport port-security aging time 34<br>switchport port-security violation shutdown<br>switchport port-security mac-address 3456.3456.5678<br>spanning-tree bpdufilter enable<br>mac access-group ISC-FastEthernet3/23 in<br>!<br>**mac access-list extended ISC-FastEthernet3/31**<br>**deny any host 0100.0ccc.cccc**<br>**deny any host 0100.0ccc.cccd**<br>**deny any host 0100.0ccd.cdd0**<br>**deny any host 0180.c200.0000**<br>**deny any host 1234.3234.3432**<br>**permit any any** | **interface GigabitEtherne4/0/1**<br>**no shut**<br>**service instance 10 ethernet**<br>**encapsulation dot1q 500**<br>**rewrite ingress tag push dot1q 555 symmetric**<br>**xconnect 192.169.105.20 505 encapsulation mpls** |

## Comments

- UNI on U-PE.
- Single match tag is performed.
- The rewrite operation **push** pushes the outer VLAN tag of 555.

# FlexUNI/EVC (Pseudowire Core Connectivity, UNI, without Port Security, with Bridge Domain)

## Configuration

- Service: FlexUNI(EVC)/Metro Ethernet.
- Feature: FlexUNI/EVC with pseudowire core connectivity, with UNI, without port security, and with bridge domain.
- Device configuration:
  - The N-PE is a Cisco 7600 with IOS 12.2(33)SRB3.

    Interface(s): GI2/0/0.
  - The U-PE is a Cisco 3750ME with IOS 12.2(25)EY2. Port security is enabled.

    Interface(s): FA1/14– FA3/23.

## Configlets

| U-PE | N-PE |
|---|---|
| vlan 772<br>exit<br>!<br>interface FastEthernet3/23<br>**switchport trunk allowed vlan 500,772**<br>!<br>interface FastEthernet1/14<br>no cdp enable<br>no keepalive<br>no ip address<br>switchport trunk allowed vlan 500,772<br>spanning-tree bpdufilter enable<br>mac access-group ISC-FastEthernet3/23 in<br>!<br>**mac access-list extended**<br>**ISC-FastEthernet1/14**<br>**deny any host 0100.0ccc.cccc**<br>**deny any host 0100.0ccc.cccd**<br>**deny any host 0100.0ccd.cdd0**<br>**deny any host 0180.c200.0000**<br>**permit any any** | **vlan 100**<br>**interface GigabitEtherne2/0/0**<br>**no shut**<br>**service instance 10 ethernet**<br>**encapsulation dot1q 500**<br>**rewrite ingress tag  push dot1q 23**<br>**second-dot1q 41 symmetric**<br>**bridge-domain 100 split-horizon**<br><br>**Interface Vlan100**<br>**no shut**<br>**xconnect 192.169.105.20 101 encapsulation**<br>**mpls** |

## Comments

- UNI on U-PE.
- Single match tag is performed.
- The rewrite operation **push** pushes two tags.

# FlexUNI/EVC (Pseudowire Core Connectivity, UNI, and Pseudowire Tunneling)

## Configuration

- Service: FlexUNI(EVC)/Metro Ethernet.
- Feature: FlexUNI/EVC with pseudowire core connectivity, with UNI, with pseudowire tunneling.
- Device configuration:
  - The N-PE is a Cisco 7600 with IOS 12.2(33) SRB3.

    Interface(s): GI4/0/0 <–> GI2/0/0.

## Configlets

| U-PE | N-PE |
|---|---|
| (None) | pseudowire-class ISC-pw-tunnel-2147<br>encapsulation mpls<br>preferred-path interface Tunnel2147<br>disable-fallback<br><br>interface GigabitEtherne4/0/0<br>service instance 1 ethernet<br>encapsulation dot1q 11 second-dot1q 41<br>rewrite ingress tag pop 2 symmetric<br>xconnect pw-class ISC-pw-tunnel-2147 |

## Comments

- UNI on N-PE (the CE is directly connected).
- Match of both tags is performed.
- The rewrite operation pops both the inner and outer VLAN tags.

# FlexUNI/EVC (VPLS Core Connectivity, UNI Port Security)

## Configuration

- Service: FlexUNI(EVC)/Metro Ethernet.
- Feature: FlexUNI/EVC with VPLS core connectivity, with UNI port security.
- Device configuration:
  - The N-PE is a Cisco 7600 with IOS 12.2(33) SRB3.

    Interface(s): GI4/0/1.
  - The U-PE is a Cisco 3750ME with IOS 12.2(25) EY2. Port security is enabled.

    Interface(s): FA1/14– FA3/23.

## Configlets

| U-PE | N-PE |
|---|---|
| vlan 788<br>exit<br>!<br>interface FastEthernet3/23<br>no ip address<br>switchport trunk allowed vlan 783,787-788<br>!<br>interface FastEthernet1/14<br>no cdp enable<br>no keepalive<br>no ip address<br>switchport<br>switchport trunk encapsulation dot1q<br>switchport mode trunk<br>switchport trunk allowed vlan none<br>**switchport trunk allowed vlan 788**<br>switchport port-security<br>switchport nonegotiate<br>switchport port-security maximum 58<br>switchport port-security aging time 85<br>switchport port-security violation shutdown<br>switchport port-security mac-address 1252.1254.2544<br>spanning-tree bpdufilter enable<br>mac access-group ISC-FastEthernet3/23 in<br>!<br>**mac access-list extended ISC-FastEthernet3/31**<br>**deny any host 0100.0ccc.cccc**<br>**deny any host 0100.0ccc.cccd**<br>**deny any host 0100.0ccd.cdd0**<br>**deny any host 0180.c200.0000**<br>**deny any host 1234.3234.3432**<br>permit any any | **l2 vfi attest-226 manual**<br>**vpn id 226**<br>**neighbor 192.169.105.20 encapsulation mpls**<br><br>**vlan 200**<br>**bridge-domain 200 split-horizon**<br><br>**interface GigabitEtherne4/0/1**<br>**no shut**<br>**service instance 10 ethernet**<br>**encapsulation dot1q 500**<br>**rewrite ingress tag translate 1-to-1 dot1q 222 symmetric**<br><br>**Interface vlan 200**<br>**xconnect vfi attest-226** |

## Comments

- UNI on U-PE.
- The rewrite operation translates the incoming VLAN tag 500 to 222.

# FlexUNI/EVC (VPLS Core Connectivity, no UNI Port Security)

## Configuration

- Service: FlexUNI(EVC)/Metro Ethernet.
- Feature: FlexUNI/EVC with VPLS core connectivity, without UNI port security.
- Device configuration:
  - The N-PE is a Cisco 7600 with IOS 12.2(33) SRB3.

    Interface(s): GI4/0/1.
  - The U-PE is a Cisco 3750ME with IOS 12.2(25) EY2.

    Interface(s): FA1/14– FA3/23.

## Configlets

| U-PE | N-PE |
|---|---|
| `vlan 772`<br>`exit`<br>`!`<br>`interface FastEthernet3/23`<br>`switchport trunk allowed vlan 500,772`<br>`!`<br>`interface FastEthernet1/14`<br>`no cdp enable`<br>`no keepalive`<br>`no ip address`<br>`switchport trunk allowed vlan 500,772`<br>`spanning-tree bpdufilter enable`<br>`mac access-group ISC-FastEthernet3/23 in`<br>`!`<br>**`mac access-list extended`**<br>**`ISC-FastEthernet1/14`**<br>**`deny any host 0100.0ccc.cccc`**<br>**`deny any host 0100.0ccc.cccd`**<br>**`deny any host 0100.0ccd.cdd0`**<br>**`deny any host 0180.c200.0000`**<br>`permit any any` | **`l2 vfi attest1-458 manual`**<br>**`vpn id 452`**<br>**`neighbor 192.169.105.20 encapsulation mpls`**<br><br>**`vlan 200`**<br>**`bridge-domain 200 split-horizon`**<br><br>**`interface GigabitEtherne4/0/1`**<br>**`no shut`**<br>**`service instance 10 ethernet`**<br>**`encapsulation dot1q 500`**<br>**`rewrite ingress tag translate 1-to-2 dot1q`**<br>**`222 second-dot1q 41 symmetric`**<br><br>**`Interface vlan 200`**<br>**`xconnect vfi attest1-458`** |

## Comments

- UNI on U-PE.
- The rewrite operation translates the incoming VLAN tag 500 to two tags, 222 and 41.

# FlexUNI/EVC (Local Connect Core Connectivity, UNI Port Security)

## Configuration

- Service: FlexUNI(EVC)/Metro Ethernet.
- Feature: FlexUNI/EVC with local connect core connectivity, with UNI port security.
- Device configuration:
  - The N-PE is a Cisco 7600 with IOS 12.2(33) SRB3.
    Interface(s):GI2/0/0.
  - The U-PE is a Cisco 3750ME with IOS 12.2(25) EY2. Port security is enabled.
    Interface(s): FA1/14– FA3/23.

## Configlets

| U-PE | N-PE |
|---|---|
| vlan 788<br>exit<br>!<br>interface FastEthernet3/23<br>no ip address<br>switchport trunk allowed vlan 783,787-788<br>!<br>interface FastEthernet1/14<br>no cdp enable<br>no keepalive<br>no ip address<br>**switchport**<br>**switchport trunk encapsulation dot1q**<br>**switchport mode trunk**<br>switchport trunk allowed vlan none<br>switchport trunk allowed vlan 788<br>switchport port-security<br>switchport nonegotiate<br>switchport port-security maximum 45<br>switchport port-security aging time 34<br>switchport port-security violation shutdown<br>switchport port-security mac-address 4111.4545.1211<br>spanning-tree bpdufilter enable<br>mac access-group ISC-FastEthernet3/23 in<br>!<br>**mac access-list extended ISC-FastEthernet3/31**<br>**deny any host 0100.0ccc.cccc**<br>**deny any host 0100.0ccc.cccd**<br>**deny any host 0100.0ccd.cdd0**<br>**deny any host 0180.c200.0000**<br>**deny any host 1234.3234.3432**<br>**permit any any** | **Connect Customer_1 GigabitEthernet4/0/1 10 GigabitEthernet4/0/10 25**<br><br>**interface GigabitEtherne4/0/1**<br>**no shut**<br>**service instance 10 ethernet**<br>**encapsulation dot1q 500**<br>**rewrite ingress tag push dot1q 555 symmetric**<br><br>**interface GigabitEtherne4/0/10**<br>**no shut**<br>**service instance 25 ethernet**<br>**encapsulation dot1q 500 second-dot1q 501**<br>**rewrite ingress tag translate 2-to-1 dot1q 222 symmetric** |

## Comments

- UNI on U-PE.
- Two tag matching operations are carried out.

- The rewrite operation translates two tags to a single tag.
- Two service instances are connected through the **connect** command.

# FlexUNI/EVC (Local Connect Core Connectivity, UNI, no Port Security, Bridge Domain)

## Configuration

- FlexUNI(EVC)/Metro Ethernet.
- Feature: FlexUNI/EVC with local connect core connectivity, with UNI, without port security, and with bridge domain.
- Device configuration:
  - The N-PE is a Cisco 7600 with IOS 12.2(33) SRB3.

    Interface(s):GI2/0/0.
  - The U-PE is a Cisco 3750ME with IOS 12.2(25) EY2.

    Interface(s):FA1/14– FA3/23.

## Configlets

| U-PE | N-PE |
|---|---|
| vlan 772<br>exit<br>!<br>interface FastEthernet3/23<br>switchport trunk allowed vlan 500,772<br>!<br>interface FastEthernet1/14<br>no cdp enable<br>no keepalive<br>no ip address<br>switchport trunk allowed vlan 500,772<br>spanning-tree bpdufilter enable<br>mac access-group ISC-FastEthernet3/23 in<br>!<br>**mac access-list extended**<br>**ISC-FastEthernet1/14**<br>**deny any host 0100.0ccc.cccc**<br>**deny any host 0100.0ccc.cccd**<br>**deny any host 0100.0ccd.cdd0**<br>**deny any host 0180.c200.0000**<br>permit any any | **interface GigabitEtherne2/0/0**<br>**no shut**<br>**service instance 10 ethernet**<br>**encapsulation dot1q 500 second-dot1q 501**<br>**rewrite ingress tag translate 2-to-2 dot1q**<br>**222 second-dot1q 41 symmetric**<br>**bridge-domain 200 split-horizon**<br><br>**interface GigabitEtherne2/0/10**<br>**no shut**<br>**service instance 15 ethernet**<br>**encapsulation dot1q 24**<br>**rewrite ingress tag pop 1 symmetric**<br>**bridge-domain 200 split-horizon** |

## Comments

- UNI on U-PE.
- The rewrite operation maps/translates the incoming two tags into two different tags.
- The service instances here are connected through bridge domain.

# FlexUNI/EVC (Pseudowire Core Connectivity, Bridge Domain, Pseudowire on SVI)

## Configuration

- FlexUNI(EVC)/Metro Ethernet.
- Feature: FlexUNI/EVC with pseudowire core connectivity, with bridge domain, and with Pseudowire on SVI enabled on the N-PE.
- Device configuration:
  - The N-PE is a Cisco 7600 with IOS 12.2(33) SRB3.

    Interface(s): GigabitEthernet7/0/0.
  - The U-PE is a Cisco 3750ME with IOS 12.2(25) EY2.

    Interface(s): FastEthernet1/0/10.

## Configlets

| U-PE | N-PE |
|---|---|
| vlan 452<br>exit<br>!<br>interface FastEthernet1/0/10<br>no ip address<br>switchport trunk allowed vlan add 452<br>!<br>interface FastEthernet1/0/13<br>no spanning-tree bpdufilter enable<br>switchport<br>no keepalive<br>no ip address<br>switchport<br>switchport trunk encapsulation dot1q<br>switchport mode trunk<br>switchport trunk allowed vlan 452<br>switchport nonegotiate | vlan 3524<br>exit<br>!<br>ethernet evc Customer1_253<br>!<br>interface GigabitEthernet7/0/0<br>**service instance 3 ethernet Customer1_253**<br>encapsulation dot1q 452<br>rewrite ingress tag pop 1 symmetric<br>**bridge-domain 3524 split-horizon**<br>!<br>**interface Vlan3524**<br>no ip address<br>description BD=T,SVI=T,Flex<br>**xconnect 22.22.22.22 52500 encapsulation mpls**<br>**backup peer 22.22.22.22 52501**<br>no shutdown |

## Comments

- None.

# FlexUNI/EVC (Pseudowire Core Connectivity, no Bridge Domain, no Pseudowire on SVI)

## Configuration

- FlexUNI(EVC)/Metro Ethernet.
- Feature: FlexUNI/EVC with pseudowire core connectivity, bridge domain disables, and with Pseudowire on SVI disabled on the N-PE.
- Device configuration:
  - The N-PE is a Cisco 7600 with IOS 12.2(33) SRB3.

    Interface(s): GigabitEthernet7/0/0.
  - The U-PE is a Cisco 3750ME with IOS 12.2(25) EY2.

    Interface(s): FastEthernet1/0/10.

## Configlets

| U-PE | N-PE |
|---|---|
| vlan 545<br>exit<br>!<br>interface FastEthernet1/0/10<br>no ip address<br>switchport trunk allowed vlan add 545<br>!<br>interface FastEthernet1/0/12<br>no spanning-tree bpdufilter enable<br>switchport<br>no keepalive<br>no ip address<br>switchport<br>switchport trunk encapsulation dot1q<br>switchport mode trunk<br>switchport trunk allowed vlan 545<br>switchport nonegotiate<br>mac access-group ISC-FastEthernet1/0/12 in | ethernet evc Customer1_248<br>!<br>interface GigabitEthernet7/0/0<br>**service instance 2 ethernet Customer1_248**<br>encapsulation dot1q 545<br>rewrite ingress tag pop 1 symmetric<br>**xconnect 22.22.22.22 52498 encapsulation mpls**<br>**backup peer 22.22.22.22 52499** |

## Comments

- None.

# FlexUNI/EVC (AutoPick Service Instance Name)

## Configuration

- FlexUNI(EVC)/Metro Ethernet.
- Feature: FlexUNI/EVC with AutoPick Service Instance Name enabled and the Service Instance Name input field left blank.
- Device configuration:
  - The N-PE is a Cisco 7600 with IOS 12.2(33) SRB3.

    Interface(s): GigabitEthernet7/0/2.
  - The U-PE is a Cisco 3750ME with IOS 12.2(25) EY2.

    Interface(s): FastEthernet1/0/14.

## Configlets

| U-PE | N-PE |
|---|---|
| `!` | `!` |
| `vlan 452` | `vlan 3524` |
| `exit` | `exit` |
| `!` | `!` |
| `interface FastEthernet1/0/10` | `ethernet evc C1_1` |
| `no ip address` | `!` |
| `switchport trunk allowed vlan add 452` | `interface GigabitEthernet7/0/0` |
| `!` | `service instance 3 ethernet C1_1` |
| `interface FastEthernet1/0/13` | `encapsulation dot1q 452` |
| `no spanning-tree bpdufilter enable` | `rewrite ingress tag pop 1 symmetric` |
| `switchport` | `bridge-domain 3524 split-horizon` |
| `no keepalive` | `!` |
| `no ip address` | `interface Vlan3524` |
| `switchport` | `no ip address` |
| `switchport trunk encapsulation dot1q` | `description BD=T,SVI=T,Flex` |
| `switchport mode trunk` | `xconnect 22.22.22.22 52500 encapsulation` |
| `switchport trunk allowed vlan 452` | `mpls` |
| `switchport nonegotiate` | `backup peer 22.22.22.22 52501` |
| `mac access-group ISC-FastEthernet1/0/13 in` | `no shutdown` |

## Comments

- The transport type is pseudowire.
- The autopick Service Instance Name will take the value *CustomerName_JobID*.

# FlexUNI/EVC (No AutoPick Service Instance Name, No Service Instance Name)

## Configuration

- FlexUNI(EVC)/Metro Ethernet.
- Feature: FlexUNI/EVC with AutoPick Service Instance Name not enabled and the Service Instance Name input field left blank.
- Device configuration:
  - The N-PE is a Cisco 7600 with IOS 12.2(33) SRB3.

    Interface(s): GigabitEthernet7/0/2.
  - The U-PE is a Cisco 3750ME with IOS 12.2(25) EY2.

    Interface(s): FastEthernet1/0/14.

## Configlets

| U-PE | N-PE |
|---|---|
| `!`<br>`vlan 566`<br>`exit`<br>`!`<br>`interface FastEthernet1/0/14`<br>`no spanning-tree bpdufilter enable`<br>`switchport`<br>`no keepalive`<br>`no ip address`<br>`switchport trunk encapsulation dot1q`<br>`switchport mode trunk`<br>`switchport trunk allowed vlan 566`<br>`switchport nonegotiate`<br>`no shutdown`<br>`mac access-group ISC-FastEthernet1/0/14 in`<br>`!`<br>`interface FastEthernet1/0/18`<br>`no ip address`<br>`switchport trunk allowed vlan 566`<br>`!`<br>`mac access-list extended`<br>`ISC-FastEthernet1/0/14`<br>`deny any host 0100.0ccc.cccc`<br>`deny any host 0100.0ccc.cccd`<br>`deny any host 0100.0ccd.cdd0`<br>`deny any host 0180.c200.0000`<br>`permit any any` | `!`<br>`interface GigabitEthernet7/0/2`<br>**`service instance 43 ethernet`**<br>`encapsulation dot1q 566`<br>`xconnect 1.1.1.1 453366 encapsulation mpls` |

## Comments

- In this example, the user does not enable AutoPick Service Instance Name and also leaves the Service Instance Name input field blank.
- The global command **ethernet evc** is not generated, while the command **service instance 43 ethernet** is generated.
- There is no Service Instance Name available and the Service Instance ID is 43.

# FlexUNI/EVC (User-Provided Service Instance Name, Pseudowire Core Connectivity)

## Configuration

- FlexUNI(EVC)/Metro Ethernet.
- Feature: FlexUNI/EVC with pseudowire core connectivity and user-provided service instance name.
- Device configuration:
  - The N-PE is a Cisco 7600 with IOS 12.2(33) SRB3.

    Interface(s): GigabitEthernet7/0/0.
  - The U-PE is a Cisco 3750ME with IOS 12.2(25) EY2.

    Interface(s): FastEthernet1/0/10.

## Configlets

| U-PE | N-PE |
|---|---|
| `!`<br>`vlan 452`<br>`exit`<br>`!`<br>`interface FastEthernet1/0/10`<br>`no ip address`<br>`switchport trunk allowed vlan add 452`<br>`!`<br>`interface FastEthernet1/0/13`<br>`no spanning-tree bpdufilter enable`<br>`switchport`<br>`no keepalive`<br>`no ip address`<br>`switchport`<br>`switchport trunk encapsulation dot1q`<br>`switchport mode trunk`<br>`switchport trunk allowed vlan 452`<br>`switchport nonegotiate`<br>`mac access-group ISC-FastEthernet1/0/13 in` | `!`<br>`vlan 3524`<br>`exit`<br>`!`<br>**`ethernet evc ServiceInst`**<br>`!`<br>`interface GigabitEthernet7/0/0`<br>**`service instance 3 ethernet  ServiceInst`**<br>`encapsulation dot1q 452`<br>`rewrite ingress tag pop 1 symmetric`<br>`bridge-domain 3524 split-horizon`<br>`!`<br>`interface Vlan3524`<br>`no ip address`<br>`description BD=T,SVI=T,Flex`<br>`xconnect 22.22.22.22 52500 encapsulation mpls`<br>`backup peer 22.22.22.22 52501`<br>`no shutdown` |

## Comments

- The transport type is PSEUDOWIRE.
- The user manually provided **ServiceInst** as the Service Instance Name. This is pushed onto the device, where the Service Instance ID is 3.

# FlexUNI/EVC (User-Provided Service Instance Name, Local Core Connectivity)

## Configuration

- FlexUNI(EVC)/Metro Ethernet.
- Feature: FlexUNI/EVC with local core connectivity and a user-provided service instance name.
- Device configuration:
  - The N-PE is a Cisco 7600 with IOS 12.2(33) SRB3.
    Interface(s): GigabitEthernet1/0/6, GigabitEthernet1/0/7.
  - The U-PE is a Cisco 3750ME with IOS 12.2(25) EY2.
    Interface(s): FastEthernet1/0/12, FastEthernet1/0/14.

## Configlets

| U-PE | N-PE |
|---|---|
| `vlan 45`<br>`exit`<br>`!`<br>`interface FastEthernet1/0/12`<br>`no ip address`<br>`switchport`<br>`switchport trunk encapsulation dot1q`<br>`switchport mode trunk`<br>`switchport trunk allowed vlan 45`<br>`!`<br>`interface FastEthernet1/0/14`<br>`no spanning-tree bpdufilter enable`<br>`switchport`<br>`no keepalive`<br>`no ip address`<br>`switchport trunk encapsulation dot1q`<br>`switchport mode trunk`<br>`switchport trunk allowed vlan 45`<br>`switchport nonegotiate`<br>`no shutdown`<br>`mac access-group ISC-FastEthernet1/0/14 in`<br>`!`<br>`mac access-list extended`<br>`ISC-FastEthernet1/0/14`<br>`deny any host 0100.0ccc.cccc`<br>`deny any host 0100.0ccc.cccd`<br>`deny any host 0100.0ccd.cdd0`<br>`deny any host 0180.c200.0000`<br>`permit any any` | **`ethernet evc  service_int`**<br>`!`<br>`interface GigabitEthernet1/0/6`<br>`no shutdown`<br>**`service instance 5 ethernet service_int`**<br>`encapsulation dot1q 56`<br>`!`<br>`interface GigabitEthernet1/0/7`<br>`no shutdown`<br>**`service instance 33 ethernet service_int`**<br>` encapsulation dot1q 45`<br>`!`<br>`connect Customer2_195 GigabitEthernet1/0/7`<br>`33 GigabitEthernet1/0/6 5` |

## Comments

- The transport type is LOCAL.
- The user manually provided **service_int** as the Service Instance Name. This is pushed onto the device, where the Service Instance IDs are 5 and 33, respectively.

# FlexUNI/EVC (User-Provided Service Instance Name, VPLS Core Connectivity)

## Configuration

- FlexUNI(EVC)/Metro Ethernet.
- Feature: FlexUNI/EVC with VPLS core connectivity and user-provided service instance name.
- Device configuration:
  - The N-PE is a Cisco 7600 with IOS 12.2(33) SRB3.

    Interface(s): GigabitEthernet7/0/0.
  - The U-PE is a Cisco 3750ME with IOS 12.2(25) EY2.

    Interface(s): FastEthernet1/0/10.

## Configlets

| U-PE | N-PE |
|---|---|
| `!`<br>`vlan 452`<br>`exit`<br>`!`<br>`interface FastEthernet1/0/10`<br>`no ip address`<br>`switchport trunk allowed vlan add 452`<br>`!`<br>`interface FastEthernet1/0/13`<br>`no spanning-tree bpdufilter enable`<br>`switchport`<br>`no keepalive`<br>`no ip address`<br>`switchport`<br>`switchport trunk encapsulation dot1q`<br>`switchport mode trunk`<br>`switchport trunk allowed vlan 452`<br>`switchport nonegotiate`<br>`mac access-group ISC-FastEthernet1/0/13 in` | `l2 vfi vpls-test manual`<br>`vpn id 300`<br>`neighbor 22.22.22.22 encapsulation mpls`<br>`!`<br>`vlan 500`<br>`!`<br>`ethernet evc ServiceInst`<br>`!`<br>`interface GigabitEtherne7/0/0`<br>`service instance 10 ethernet ServiceInst`<br>`encapsulation dot1q 400`<br>`rewrite ingress tag pop 1 symmetric`<br>`bridge-domain 500 split-horizon`<br>`!`<br>`interface vlan500`<br>`xconnect vfi vpls-test` |

## Comments

- The transport type is VPLS.
- The user manually provided **ServiceInst** as the Service Instance Name. This is pushed onto the device, where the Service Instance ID is 10.

# FlexUNI/EVC (ATM-Ethernet Interworking, Pseudowire Core Connectivity, Point-to-Point Circuit)

## Configuration

- FlexUNI(EVC)/ATM-Ethernet Interworking.
- Feature: FlexUNI/EVC for ATM-Ethernet interworking with pseudowire core connectivity with an end-to-end circuit with multiple links. One link terminates on an ATM interface on N-PE 1, and the other link terminates on an Ethernet interface on N-PE 2.
- Device configuration:
  - N-PE 1 is a Cisco 7600 with IOS 12.2(33) SRB3.

    Interface(s): ATM1/0/0.370.
  - N-PE 2 is a Cisco 7600 with IOS 12.2(33) SRE.

    Interface(s): GigabitEthernet4/0/2.

## Configlets

| N-PE 1 (ATM) | N-PE 2 (Ethernet) |
|---|---|
| `!`<br>`interface ATM1/0/0.370 point-to-point`<br>` no atm enable-ilmi-trap`<br>` pvc 0/370 l2transport`<br>`  encapsulation aal5snap`<br>`  xconnect 192.169.105.10 123 pw-class inter-ether`<br>`  !` | `!`<br>`ethernet evc 1-3_51`<br>`!`<br>`interface GigabitEthernet4/0/2`<br>` no ip address`<br>` no mls qos trust`<br>` service instance 103 ethernet 1-3_51`<br>`  encapsulation dot1q 370`<br>`  rewrite ingress tag pop 1 symmetric`<br>`  xconnect 192.169.105.20 123 encapsulation mpls`<br>`!` |

## Comments

- None.

# FlexUNI/EVC (ATM-Ethernet Interworking, Pseudowire Core Connectivity, Multipoint Circuit)

## Configuration

- FlexUNI(EVC)/ATM-Ethernet Interworking.
- Feature: FlexUNI/EVC for ATM-Ethernet interworking with pseudowire core connectivity with a multipoint circuit. Link #1 terminates on an ATM interface on N-PE 1, link #2 terminates on an Ethernet interface on N-PE 1, and link #3 terminates on an Ethernet interface on N-PE 2.
- Device configuration:
  - The N-PE 1 is a Cisco 7600 with IOS 12.2(33) SRB3.

    Interface(s): GigabitEthernet7/0/4, ATM6/0/0.100.
  - The N-PE 2 is a Cisco 7600 with IOS 12.2(33) SRE.

    Interface(s): GigabitEthernet7/0/5.

## Configlets

| N-PE 1 (ATM + Ethernet) | N-PE 2 (Ethernet) |
|---|---|
| `!` | `!` |
| `vlan 500` | `vlan 800` |
| `exit` | `exit` |
| `!` | `!` |
| `ethernet evc Customer1_166` | `ethernet evc Customer1_166` |
| `!` | `!` |
| `interface GigabitEthernet7/0/4` | `interface GigabitEthernet7/0/5` |
| `no shutdown` | `no shutdown` |
| `service instance 1 ethernet Customer1_166` | `service instance 1 ethernet Customer1_166` |
| `encapsulation dot1q 600` | `encapsulation dot1q 623` |
| `bridge-domain 500 split-horizon` | `bridge-domain 800 split-horizon` |
| `!` | `!` |
| `interface ATM6/0/0.100 point-to-point` | `interface Vlan800` |
| `pvc 200/300` | `description UT-9` |
| `encapsulation aal5snap` | `xconnect 192.169.105.20 6 pw-class` |
| `bridge-domain 500 split-horizon` | `ISC-pw-tunnel-900` |
| `!` | |
| `interface Vlan500` | |
| `no ip address` | |
| `description UT-9` | |
| `xconnect 1.1.1.1 6 pw-class` | |
| `ISC-pw-tunnel-400` | |
| `no shutdown` | |

## Comments

- None.

# FlexUNI/EVC (ATM-Ethernet Interworking, Local Core Connectivity, Point-to-Point Circuit)

## Configuration

- FlexUNI(EVC)/ATM-Ethernet Interworking.
- Feature: FlexUNI/EVC for ATM-Ethernet interworking with local core connectivity with a point-to-point circuit. The circuit terminates on different ATM interfaces on the same local N-PE.
- Device configuration:
  - The N-PE 1 is a Cisco 7600 with IOS 12.2(33) SRB3.

    Interface(s): ATM1/0/1, ATM4/1/0, ATM1/0/1.99, ATM4/1/0.98.

## Configlets

| N-PE 1 (ATM) | N/A |
|---|---|
| `!`<br>`interface ATM1/0/1`<br>`no shutdown`<br>`!`<br>`interface ATM4/1/0`<br>`no shutdown`<br>`!`<br>`interface ATM1/0/1.99 point-to-point`<br>`pvc 99/99 l2transport`<br>`encapsulation aal0`<br>`!`<br>`interface ATM4/1/0.98 point-to-point`<br>`pvc 98/98 l2transport`<br>`encapsulation aal0`<br>`!`<br>`connect ATM-to-ATM ATM1/0/1 99/99 ATM4/1/0 98/98`<br>`!` | |

## Comments

- None.

# FlexUNI/EVC (ATM-Ethernet Interworking, Local Core Connectivity, Multipoint Circuit)

## Configuration

- FlexUNI(EVC)/ATM-Ethernet Interworking.
- Feature: FlexUNI/EVC for ATM-Ethernet interworking with local core connectivity for multiple links that terminate on the same local N-PE. Link #1 terminates on an ATM interface, and link #2 terminates on an Ethernet interface.
- Device configuration:
  - The N-PE 1 is a Cisco 7600 with IOS 12.2(33) SRB3.

    Interface(s): ATM1/0/0.99, TenGigabitEthernet6/0/0, TenGigabitEthernet6/0/1.

## Configlets

<table>
<tr><th>N-PE 1 (ATM + Ethernet)</th><th>N/A</th></tr>
<tr><td>

```
!
vlan 1001
exit
!
interface ATM1/0/0.99 point-to-point
 no atm enable-ilmi-trap
 pvc 99/99
  encapsulation aal5snap
 bridge-domain 1001
 !
!
interface TenGigabitEthernet6/0/0
 no ip address
 no mls qos trust
 service instance 104 ethernet 1-4_60
  encapsulation dot1q 11
  rewrite ingress tag pop 1 symmetric
  bridge-domain 1001
 !
!
interface TenGigabitEthernet6/0/1
 no ip address
 no mls qos trust
 service instance 105 ethernet 1-4_60
  encapsulation dot1q 12
  rewrite ingress tag pop 1 symmetric
  bridge-domain 1001
 !
```

</td><td></td></tr>
</table>

## Comments

- None.

# FlexUNI/EVC (ATM-Ethernet Interworking, Local Core Connectivity, Multipoint Circuit)

## Configuration

- FlexUNI(EVC)/ATM-Ethernet Interworking.
- Feature: FlexUNI/EVC for ATM-Ethernet interworking with local core connectivity. Multiple links terminate on the same local N-PE. Link #1 terminates on an ATM interface, link #2 terminates on an ATM interface, and link #3 terminates on an ATM interface.
- Device configuration:
  - The N-PE 1 is a Cisco 7600 with IOS 12.2(33) SRB3.

    Interface(s): ATM6/0/0.100, ATM6/0/1.101, ATM6/0/2.102.

## Configlets

| N-PE 1 (ATM) | N/A |
|---|---|
| !<br>vlan 500<br>exit<br>!<br>interface ATM6/0/0.100 point-to-point<br>pvc 200/300<br>encapsulation aal5snap<br>bridge-domain 500<br>!<br>interface ATM6/0/1.101 point-to-point<br>pvc 201/301<br>encapsulation aal5snap<br>bridge-domain 500<br>!<br>interface ATM6/0/2.102 point-to-point<br>pvc 202/302<br>encapsulation aal5snap<br>bridge-domain 500<br>! | |

## Comments

- None.

# FlexUNI/EVC (ATM-Ethernet Interworking, Local Core Connectivity, Point-to-Point Circuit)

### Configuration

- FlexUNI(EVC)/ATM-Ethernet Interworking.
- Feature: FlexUNI/EVC for ATM-Ethernet interworking with local core connectivity. A point-to-point circuit terminates on different ATM interfaces on same local N-PE.
- Device configuration:
  - The N-PE 1 is a Cisco 7600 with IOS 12.2(33) SRB3.

    Interface(s): ATM1/0/0, ATM1/0/1.

### Configlets

| N-PE 1 (ATM) | N/A |
|---|---|
| `!`<br>`interface ATM1/0/0`<br>`atm pvp 33 l2transport`<br>`!`<br>`interface ATM1/0/1`<br>`atm pvp 222 l2transport`<br>`!`<br>`connect Customer1_208 ATM1/0/0 33 ATM1/0/1 222` | |

### Comments

- None.

# FlexUNI/EVC (ATM-Ethernet Interworking, Pseudowire Core Connectivity, End-to-End Circuit)

## Configuration

- FlexUNI(EVC)/ATM-Ethernet Interworking.
- Feature: FlexUNI/EVC for ATM-Ethernet interworking with pseudowire core connectivity for end-to-end circuit with multiple links. One link terminates on ATM interface on N-PE 1, and other link terminates on an Ethernet interface on N-PE 2.
- Device configuration:
  - The N-PE 1 is a Cisco 7600 with IOS 12.2(33) SRB3.

    Interface(s): ATM1/0/0.370.
  - The N-PE 2 is a Cisco ASR 9000 with IOS XR 3.9.0.

    Interface(s): GigabitEthernet0/0/0/4.458.

## Configlets

| N-PE 1 (ATM) | N-PE 2 (Ethernet) |
|---|---|
| !<br>interface ATM1/0/0.370 point-to-point<br> no atm enable-ilmi-trap<br> pvc 0/370 l2transport<br>  encapsulation aal5snap<br>  xconnect 192.169.105.10 123 pw-class inter-ether<br> ! | interface GigabitEthernet0/0/0/4.458 l2transport<br>  encapsulation dot1q 458<br>!<br>l2vpn<br> xconnect group VPNSC<br>  p2p iscind-crs-1--48856<br>   interface GigabitEthernet0/0/0/4.458<br>   neighbor 192.168.118.167 pw-id 123<br>  !<br> !<br>! |

## Comments

- None.

# FlexUNI/EVC (ATM-Ethernet Interworking, Pseudowire Core Connectivity, Multipoint Circuit)

## Configuration

- FlexUNI(EVC)/ATM-Ethernet Interworking.
- Feature: FlexUNI/EVC for ATM-Ethernet interworking with pseudowire core connectivity with an end-to-end circuit with multiple links. One link is terminating on an ATM interface on N-PE 1, and the other (non-flex) link terminates on an Ethernet interface on N-PE 2.
- Device configuration:
  - The N-PE 1 is a Cisco 7600 with IOS 12.2(33) SRB3.

    Interface(s): ATM4/1/0.8790.
  - The N-PE 2 is a Cisco 7600 with IOS 12.2(33) SRB3.

    Interface(s): GigabitEthernet4/0/17.600.

## Configlets

| N-PE 1 (ATM) | N-PE 2 (Ethernet) |
|---|---|
| `interface ATM4/1/0.8790 point-to-point`<br>`pvc 150/3454 l2transport`<br>`encapsulation aal5snap`<br>`xconnect 192.169.105.10 760 pw-class ISC-pw-tunnel-1` | `interface GigabitEthernet4/0/17.600`<br>`encapsulation dot1Q 600`<br>`xconnect 192.169.105.20 760 pw-class ISC-pw-tunnel-1` |

## Comments

- None.

# FlexUNI/EVC (ATM-Ethernet Interworking, Local Core Connectivity, Point-to-Point Circuit)

## Configuration

- FlexUNI(EVC)/ATM-Ethernet Interworking.
- Feature: FlexUNI/EVC for ATM-Ethernet interworking with local core connectivity for point-to-point circuit. The circuit terminates on the same, local N-PE 1. One link terminates on an ATM interface, and the other (non-flex) link terminates on an Ethernet interface.
- Device configuration:
  - The N-PE 1 is a Cisco 7600 with IOS 12.2(33) SRB3.

    Interface(s): ATM1/0/0.444.
  - The N-PE 1 is a Cisco 7600 with IOS 12.2(33) SRB3.

    Interface(s): FastEthernet3/39.674.

## Configlets

| N-PE 1 (ATM + Ethernet) | N/A |
|---|---|
| !<br>interface FastEthernet3/39.674<br>encapsulation dot1Q 674<br>!<br>interface ATM1/0/0.444 point-to-point<br>pvc 44/4444 l2transport<br>encapsulation aal5snap<br>!<br>connect Customer1_204 ATM1/0/0 44/4444<br>FastEthernet3/39.674 interworking ethernet | |

## Comments

- None.

# FlexUNI/EVC (ATM-Ethernet Interworking, Pseudowire Core Connectivity, End-to-End Circuit, with Bridge Domain)

## Configuration

- FlexUNI(EVC)/ATM-Ethernet Interworking.
- Feature: FlexUNI/EVC for ATM-Ethernet interworking with pseudowire core connectivity for end-to-end circuit with multiple links with bridge domain enabled. One link terminates on an ATM interface on N-PE 1, and the other link terminates on a flex Ethernet interface on N-PE 2.
- Device configuration:
  - The N-PE 1 is a Cisco 7600 with IOS 12.2(33) SRB3.

    Interface(s): ATM1/0/0.370.
  - The N-PE 2 is a Cisco ASR 9000 with IOS XR 3.9.0.

    Interface(s): GigabitEthernet0/0/0/25.341.

## Configlets

| N-PE 1 (ATM) | N-PE 2 (Ethernet) |
|---|---|
| !<br>interface ATM1/0/0.370 point-to-point<br> no atm enable-ilmi-trap<br> pvc 0/370 l2transport<br>  encapsulation aal5snap<br>xconnect 10.20.21.1 4531 pw-class<br>ISC-pw-tunnel-1 | interface GigabitEthernet0/0/0/25.341<br>l2transport<br> encapsulation dot1q 341<br> rewrite ingress tag push dot1q 430<br>second-dot1q 349 symmetric<br>!<br>l2vpn<br> bridge group tml<br>  bridge-domain CISCO<br>   interface GigabitEthernet0/0/0/25.341<br>   !<br>   neighbor 192.169.105.20 pw-id 32190<br>   !<br>  !<br> !<br>! |

## Comments

- None.

# FlexUNI/EVC (ATM-Ethernet Interworking, Pseudowire Core Connectivity, End-to-End Circuit, with Bridge Domain)

## Configuration

- FlexUNI(EVC)/ATM-Ethernet Interworking.
- Feature: FlexUNI/EVC for ATM-Ethernet interworking with pseudowire core connectivity for end-to-end circuit with multiple links. Bridge domain is enabled. One link terminates on an ATM interface on N-PE 1, and the other (non-flex) link terminates on an Ethernet interface on N-PE 2.
- Device configuration:
  - The N-PE 1 is a Cisco 7600 with IOS 12.2(33) SRB3.

    Interface(s): ATM1/0/0.370.
  - The N-PE 2 is a Cisco ASR 9000 with IOS XR 3.9.0.

    Interface(s): GigabitEthernet0/0/0/20.712.

## Configlets

| N-PE 1 (ATM) | N-PE 2 (Ethernet) |
|---|---|
| `!`<br>`interface ATM1/0/0.370 point-to-point`<br>` no atm enable-ilmi-trap`<br>` pvc 0/370 l2transport`<br>`  encapsulation aal5snap`<br>`  xconnect 10.20.21.1 4531 pw-class`<br>`ISC-pw-tunnel-1`<br>` !` | `interface GigabitEthernet0/0/0/20.712`<br>`l2transport`<br>` encapsulation dot1q 712`<br>`!`<br>`l2vpn`<br>` bridge group tml`<br>`  bridge-domain CISCO`<br>`   interface GigabitEthernet0/0/0/20.712`<br>`   !`<br>`   neighbor 192.169.105.20 pw-id 1005`<br>`   !`<br>`  !`<br>` !`<br>`!` |

## Comments

- None.

# FlexUNI/EVC (ATM-Ethernet Interworking, Pseudowire Core Connectivity, End-to-End Circuit, no Bridge Domain)

### Configuration

- FlexUNI(EVC)/ATM-Ethernet Interworking.
- Feature: FlexUNI/EVC for ATM-Ethernet interworking with pseudowire core connectivity for end-to-end circuit with multiple links. Bridge domain is disabled. One link is terminates on an ATM interface on N-PE 1, and the other link terminates on an Ethernet interface on N-PE 2.
- Device configuration:
  - The N-PE 1 is a Cisco 7600 with IOS 12.2(33) SRB3.

    Interface(s): ATM1/0/0.370.
  - The N-PE 2 is a Cisco ASR 9000 with IOS XR 3.9.0.

    Interface(s): GigabitEthernet0/0/0/12.433.

### Configlets

| N-PE 1 (ATM) | N-PE 2 (Ethernet) |
|---|---|
| `!`<br>`interface ATM1/0/0.370 point-to-point`<br>` no atm enable-ilmi-trap`<br>` pvc 0/370 l2transport`<br>`  encapsulation aal5snap`<br>`xconnect 10.20.21.1 4531 pw-class`<br>`ISC-pw-tunnel-1 !` | `interface GigabitEthernet0/0/0/12.433`<br>`l2transport`<br>` encapsulation dot1q 433`<br>` rewrite ingress tag push dot1q 43`<br>`second-dot1q 53 symmetric`<br>`!`<br>`l2vpn`<br>` xconnect group ISC`<br>`  p2p CISCO`<br>`   interface GigabitEthernet0/0/0/12.433`<br>`   neighbor 192.169.105.20 pw-id 4531`<br>`   !`<br>`  !`<br>` !`<br>`!` |

### Comments

- None.
-

APPENDIX B

# Working with Templates and Data Files

This appendix describes how to use templates and data files with ISC policies and service requests. It contains the following sections:

- Overview, page B-1
- Using Templates with ISC Policies, page B-4
- Using Templates with Service Requests, page B-10

**Note** For an overview of the Template Manager and how templates and data files are created in ISC, see Chapter 6, “Service Design” and Appendix D, “Template Usage” in the *Cisco IP Solution Center Infrastructure Reference, 6.0.*

## Overview

ISC’s Template Manager allows you to create, manage, and store templates and associated data files. The purpose of using templates is to provide a means to download free-form CLIs to a device, in order to deploy commands and configurations not normally supported by ISC. Templates are written in the Velocity Template Language and are generally comprised of IOS (and IOS XR) device CLI configurations. Optionally, you can set variables (also defined in Velocity) that are substituted with data stored in data files. The template and data file information is downloaded to the device at the time of service activation. The information in this appendix assumes the templates (including subtemplates, if applicable) and (optional) data files have been set up.

**Note** You can also create data files “on demand” during service request creation, as covered in later sections of this appendix.

This overview section covers the following topics:

- Summary of Template Manager Features, page B-1
- Template and Data File Workflow, page B-3

### Summary of Template Manager Features

This section highlights key features of template and data file support in ISC, especially those that have an impact on working with policies and service requests.

### Template Attributes

The ISC template mechanism allows you to differentiate templates by specifying (optional) attributes on a template, including:

- Device type
- Line card type
- Port type
- Software version (IOS or IOS XR)

These attributes are set through a drop-down list when setting up the template in Template Manager. ISC uses these attributes to automatically select the template/data file that most closely matches the device defined within the service request. See the section Using Templates with Service Requests, page B-10, for additional information.

### Associating Templates at the Policy Level

ISC supports the association of templates and data files in policies.

### Selective Determination of Templates for U-PE and PE-AGG Device Roles

For added flexibility, ISC allows you to selectively apply templates to U-PE and PE-AGG devices (for example, in a ring environment) based on whether the devices have a UNI interface.

### Enhanced Subtemplate Support

A new attribute in the Template Editor allows subtemplates to be associated with a template. ISC supports dynamic instantiation of subtemplates based on device attributes. While creating the subtemplates, values for these identifiers must be provided by the operator.

### Dynamic Data File Creation

The user can create a data file during service request creation and associate it to the template copied from the associated policy. This functionality extends data file creation from the Template wizard to doing so directly from the service request wizard Template Association screen. In addition, you can modify any or all variables that are part of the template/date file attached to a service request and apply the updated template/data file without removing the entire service.

### Automatic Application of Negate Templates

To remove a configuration created from a template/data file, a negate template must be applied to the existing service. This is no longer a manual process in ISC. You create both the positive and negate template. You can assign a positive template/data file to a policy. ISC calls the appropriate negate template at the appropriate time, as the negate template has a direct relationship with the deploy template. ISC determines which negate template to use, based on the service request action requested (for example, deploying or decommissioning a service). The negate template has the same name as the template, with the addition of the suffix .Negate. The negate template does not share the data file of the deploy template. The negate template must have its own data file defined.

### Compatibility of the Template Mechanism with Previous ISC Releases

ISC maintains compatibility with the template mechanism in previous ISC releases. Templates created in earlier versions of ISC work “as is,” without any modifications to the templates or the workflow. In the case of a policy in the system that was created in an earlier ISC release, the GUI workflow for associating templates/data files is not visible. In such a case, the operator adds the template and data files during service deployment, as in previous releases of ISC.

### Template Support for IOS and IOS XR

The template mechanism is supported for both IOS and IOS XR devices. For IOS XR devices, the configlet generated from templates/data files contains CLI commands and not XML statements. For IOS XR devices, template support is provided as CLI commands. For IOS devices, the operator can download a template configlet using the device console.

**Note** Note the following known issue in the case of IOS XR devices. When a service request is deployed with templates that contain improper or unsupported configurations, the service request still goes to the DEPLOYED state. This because the IOS XR device does not issue an error report on the improper configuration(s) deployed.

### RBAC Support for Template Usage

Templates and data files are only accessible to users with the proper RBAC role. A permission type for data files has been added. The permissions allowed for the data files are view, create, modify, and delete. Operators cannot view templates/data files assigned to other roles, and are not permitted to deploy templates/data files to which they do not have access. See the *Cisco IP Solution Center Infrastructure Reference, 6.0* for more information on RBAC support for templates/data files.

### Template Variables

Template variables support most ISC repository variables for MPLS, L2VPN, VPLS, and FlexUNI/EVC. For a list of supported template variables, see the "Devices" chapter of the *Cisco IP Solution Center Infrastructure Reference, 6.0.*

### DCPL Properties

There are a few Dynamic Component Properties Library (DCPL) properties governing templates. These DCPL properties affect when a template is applied, whether negate templates are appended or prepended, whether templates are applied in the case when an service has multiple lines, only one of which have been edited, etc. For documentation on DPLC properties related to templates, see the "Properties" chapter of the *Cisco IP Solution Center Infrastructure Reference, 6.0.*

### Importing and Exporting Templates

ISC provides a mechanism to import and export templates and data files. See the *Cisco IP Solution Center Infrastructure Reference, 6.0* for more information.

## Template and Data File Workflow

This section summarizes the basic operations involved in setting up and using templates, data files, and negate templates in ISC.

### Basic Template Manager Functions

- Create templates and negate templates for different configurations.
- Specify device attributes for the templates.
- Associate subtemplates to templates, if applicable
- Create data files for the subtemplates.
- Create a negate template for each subtemplate.
- Create data files for the negate templates.

- Create a super template and attach subtemplates to it.

These basic Template Manager functions are documented in the *Cisco IP Solution Center Infrastructure Reference, 6.0*. See that guide for more information on these tasks.

### Policy-Level Template Functions

- Create a policy and enable template support for the policy.
- Associate templates and (optionally) data files to the policy, if desired.

For information on how to associate templates and data files at the policy level, see the section Using Templates with ISC Policies, page B-4, in this appendix.

### Service Request-Level Template Functions

**Note** When a policy is only associated with a template and no data file, then during creation of a service request using that policy, automatic selection of a data file for that template takes place, if the template has only one data file. If the template does not have a data file, then one must be created for that template and associated to the service request before saving is permitted.

- Create a service request and associate template(s) to a link.
- Deploy the service request on a device (for example, a 7600).
- The subtemplate and corresponding data file for the 7600 are autoselected for deployment.
- A configlet is generated from the subtemplate.
- Decommission the service request.
- The negate template for the subtemplate is autoselected and deployed.

For information on how to use templates and data files is service requests, see the section Using Templates and Data Files with Service Requests, page B-14, in this appendix.

# Using Templates with ISC Policies

This section provides information on how to enable template support and associate templates/data files with ISC policies. It contains the following sections:

- Overview of Template Support in ISC Policies, page B-4
- Associating Templates and Data Files to a Policy, page B-5

## Overview of Template Support in ISC Policies

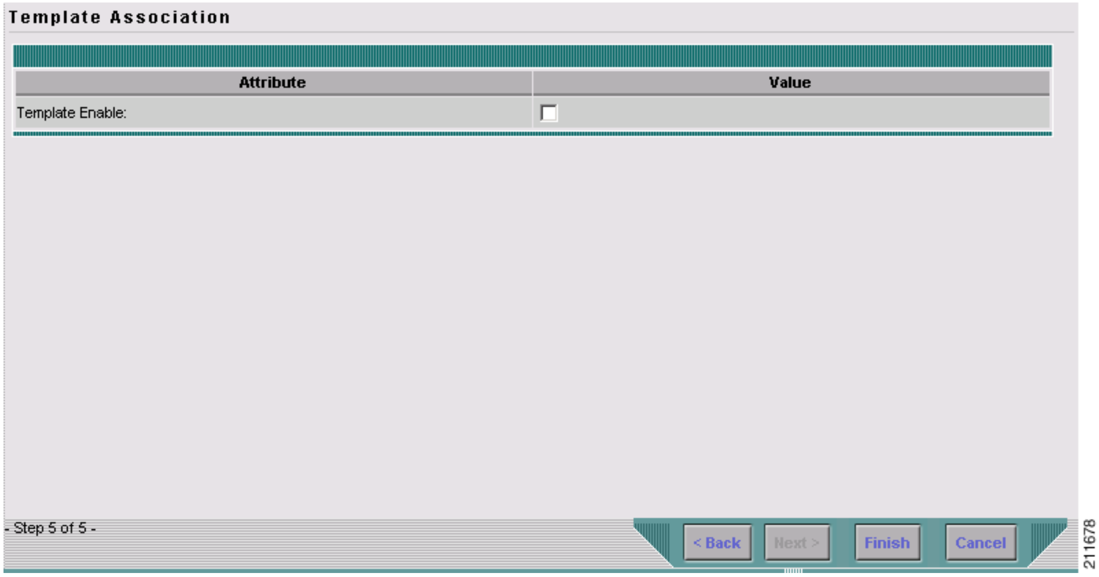
ISC supports associating templates/data files to a service policy. This minimizes steps in the provisioning workflow and also reduces potential errors that can occur if an incorrect template/data file is selected during service creation. In the Policy Editor workflow, after the policy attributes are set, a new Templates Association window appears. The Enable Templates check box that appears in this window allows you to enable template association for the policy and to specify templates/data files to be available for service requests based on the policy. More than one template/data file can be associated to the policy. Each template/data file can be associated to a device role. The available device roles are determined by the policy type. In the case of U-PE and PE-AGG device roles, templates/data files can be selectively determined based on whether the device has a UNI interface. Later, at the time of service request creation, templates are only available if the device type matches the role type specified for the template within the policy or role type along with (or without) the presence of UNI interface in the policy.

## Associating Templates and Data Files to a Policy

This section describes how to associate templates and data files to an ISC policy. These features also apply in the case of editing a policy.

After the policy attributes are set for a policy, the Template Association window appears in the workflow, as shown in Figure B-1.

*Figure B-1 Template Association Window*



Template Association

| Attribute | Value |
|---|---|
| Template Enable: | □ |

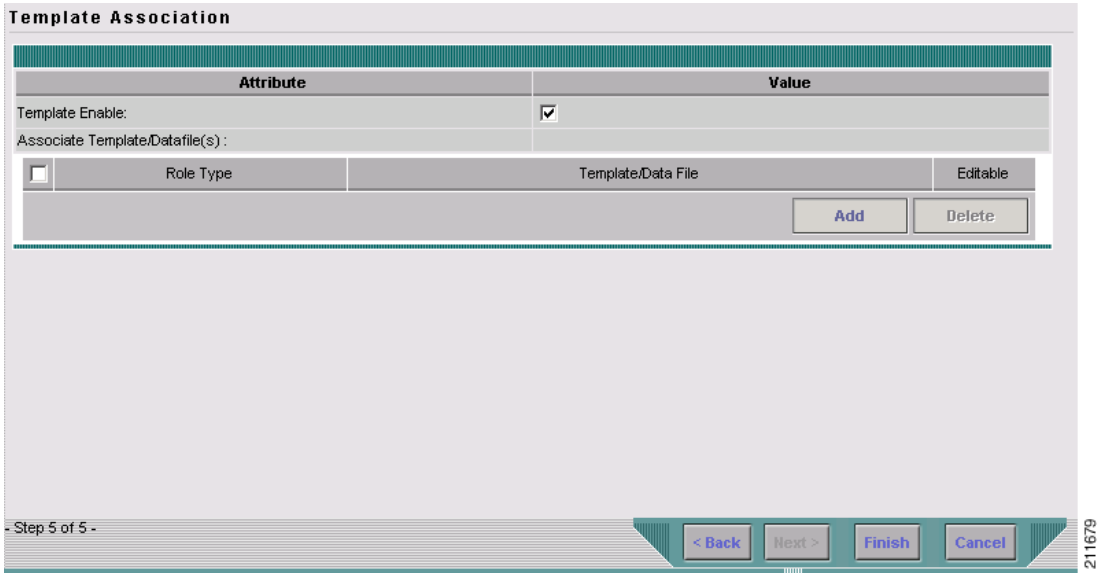
This window is where you associate the templates/data files as a final step before clicking the Finish button and saving the policy settings.

Perform the following steps to associate template(s)/data file(s) with the policy.

**Step 1** Check the **Template Enable** check box to enable template use in service requests based on this policy. This check box is unchecked by default.

The GUI updates with fields allowing you to associate templates/data files to the policy, as shown in Figure B-2.

*Figure B-2      Template Association Window with Template Enable Checked*



Template Association

| Attribute | Value |
|---|---|
| Template Enable: | ☑ |
| Associate Template/Datafile(s) : | |

| □ | Role Type | Template/Data File | Editable |
|---|---|---|---|

**Step 2** Click the **Add** button to add a row in which to specify associated templates/data files.

A new row appears in the GUI, providing fields to set the role type, specify templates/data files, and specify if the template/data file is editable within service requests based on the policy.

**Step 3** In the Role Type column, choose a device role from the drop-down list.

The role selections might include:

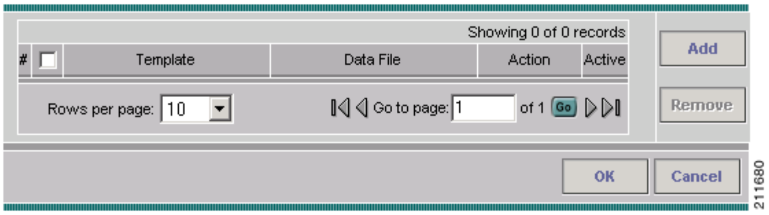
- N-PE
- PE-AGG
- U-PE
- CE (MULTI_VRF)
- CE (MANAGED)
- MVRF

**Note** The available device roles in the drop-down list are determined by the policy type.

**Step 4** To add a template/data file click the **Add** link in the Template/Data File column.

The Add/Remove Templates window appears, as shown in Figure B-3.

*Figure B-3      Add/Remove Template Pop-Up*



Showing 0 of 0 records

| # | □ | Template | Data File | Action | Active |
|---|---|---|---|---|---|

Rows per page: 10

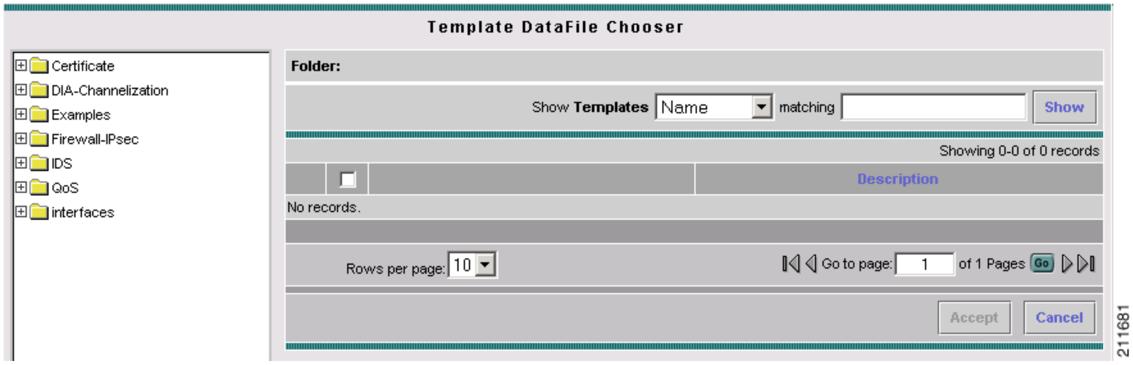
Go to page: 1 of 1

**Step 5** Click the **Add** button to select a template/data file to associate with the policy.

**Note** If the device role is specified as U-PE or PE-AGG, templates can be selectively added based on whether the device has a UNI interface. For details on this feature, see Selectively Determining Templates for U-PE and PE-AGG Device Roles, page B-9. The actual steps for adding templates/data files are the same as in the following steps.

The Template Datafile Chooser window appears, as shown in Figure B-4.

*Figure B-4 Template Datafile Chooser Window*



Template DataFile Chooser

- Certificate
- DIA-Channelization
- Examples
- Firewall-IPsec
- IDS
- QoS
- interfaces

Folder:

Show **Templates** Name matching Show

Showing 0-0 of 0 records

□ Description

No records.

Rows per page: 10 Go to page: 1 of 1 Pages Go

Accept Cancel

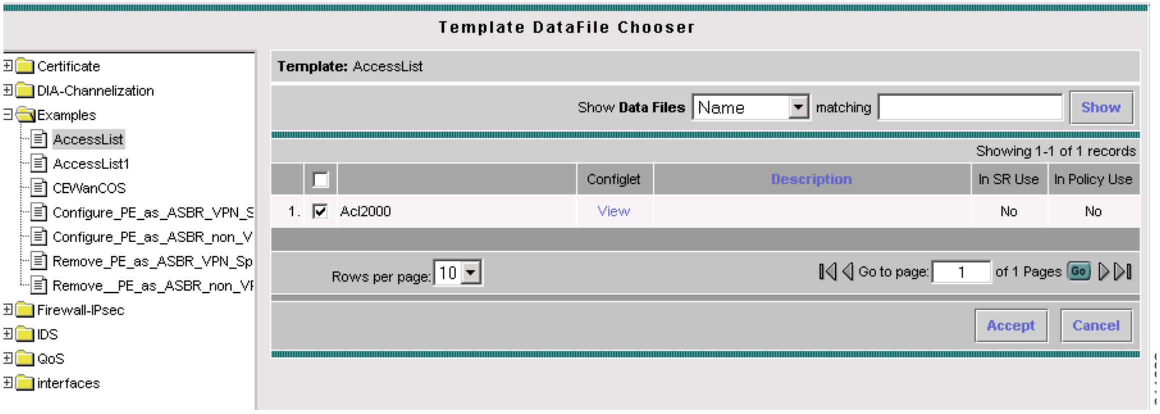
This is a standard Template Manager window used to navigate to and choose templates and (optionally) data files in ISC.

**Note** The following steps involving the Template Datafile Chooser window assume a familiarity with the functionality of the window. For additional information about Template Manager and how templates and data files are created and managed in ISC, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*. The steps shown here are for example purposes. You must modify the steps as required for your environment. For example, you might want to choose only a template file or both a template file and a data file to associate with the policy. Both scenarios are supported.

**Step 6** Navigate to a template in the folder tree and click it to select it.

The template is listed in the right side of the GUI, along with any data files that are associated with it, as shown in Figure B-5.

*Figure B-5 Template Datafile Chooser Window with Template and Data File Listed*



**Template:** AccessList

Show **Data Files** Name matching

Showing 1-1 of 1 records

| | □ | | Configlet | Description | In SR Use | In Policy Use |
|---|---|---|---|---|---|---|
| 1. | ☑ | Acl2000 | View | | No | No |

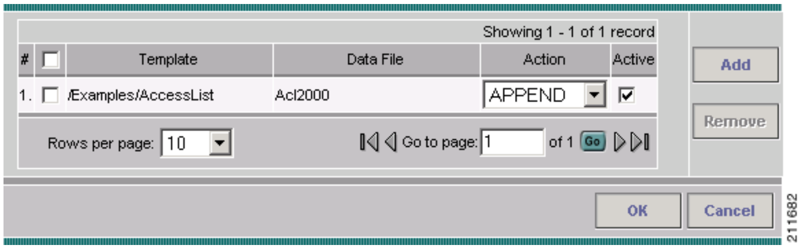
Rows per page: 10 Go to page: 1 of 1 Pages

**Step 7** Check the check box to the left of a data file name and click the **Accept** button.

**Note** You can select only the template or both template and data file at this stage, depending on your needs, and whether or not a data file exists for the template.

The Template Datafile Chooser window closes and the selected template/data file appears listed in the Add/Remove Templates window, as show in Figure B-6.

*Figure B-6 Add/Remove Templates Window with Template Listed*



Showing 1 - 1 of 1 record

| # | □ | Template | Data File | Action | Active |
|---|---|---|---|---|---|
| 1. | □ | /Examples/AccessList | Acl2000 | APPEND | ☑ |

Rows per page: 10 Go to page: 1 of 1

If you did not choose a data file, then the Datafile column is blank.

**Step 8** Check the check box to the left of the template name to choose the template.

**Step 9** Under Action, use the drop-down list and choose **APPEND** or **PREPEND.**

Append tells ISC to append the template-generated CLIs to the regular ISC (non-template) CLIs (configlet). Prepend is the reverse (adds the template to the beginning of the configlet).

**Step 10** Choose **Active** to use this template for service requests based on this policy.

If you do not choose Active, the template is not used.

**Step 11** To associate additional templates/data files with the policy click **Add** in the Add/Remove Templates window and repeat the appropriate steps to add other templates/data files.

**Step 12** To remove a template row from the window, check a template and click the **Remove** button to remove the template from the list.

**Step 13** When you are satisfied with the selections in the Add/Remove Templates window, click **OK**.

The Template Association window appears with the template(s)/data file(s) listed as active link(s). If you have added more than one template/data file, they appear in a comma-separated list of links.

You can click on any link to return to the Add/Remove Templates window, in order to edit/update the template/data file information.

**Step 14** Check the **Edit** check box to make the template/data file attributes editable in service requests based on the policy.

**Step 15** To add additional templates/data files for a given role to the policy, you can click the **Add** button in the Template Association window and repeat the steps outlined above.

**Step 16** To delete templates/data files that have been associated to the policy, check a template/data file to choose it.

Then click the **Delete** button to delete it from the Template Association window.

**Step 17** When you are finished associating the template(s)/data file(s) to the policy, click the **Finish** button in the Template Association window.

The attributes for the policy are saved and the policy creation or modification is complete.

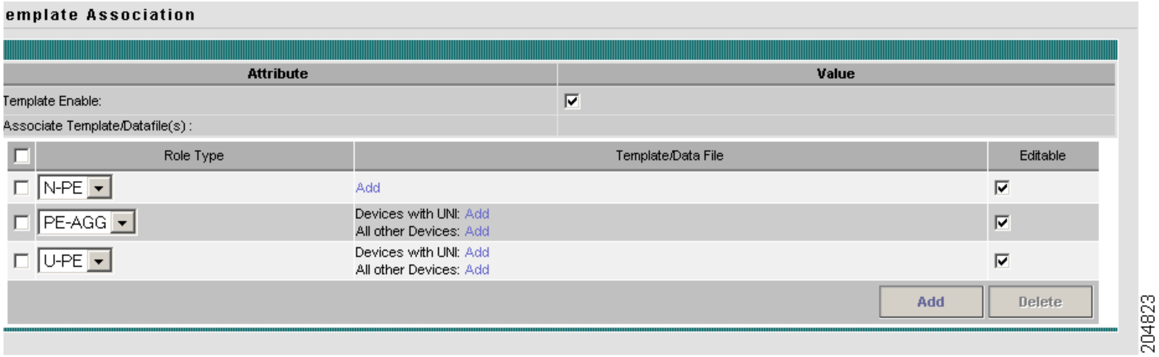
## Selectively Determining Templates for U-PE and PE-AGG Device Roles

ISC provides the capability to selectively determine which U-PE and PE-AGG devices (for example, in a ring environment) to apply templates/data files. During template association in the service policy workflow, the U-PE and PE-AGG device roles have two options to associate templates/data files. These options are:

- Devices with UNI. This option causes templates/data files to be configured on devices of the specified role with a UNI interface.
- All other devices. This option causes templates/data files to be configured on all devices of the specified role, including those with a UNI interface.

Figure B-7 shows these options in the Template/Data File column of the Template Association window.

***Figure B-7 Options for Selectively Determining Templates for U-PE and PE-AGG Device Roles***



emplate Association

| Attribute | Value |
|---|---|
| Template Enable: | ☑ |
| Associate Template/Datafile(s) : | |

| □ | Role Type | Template/Data File | Editable |
|---|---|---|---|
| □ | N-PE | Add | ☑ |
| □ | PE-AGG | Devices with UNI: Add<br>All other Devices: Add | ☑ |
| □ | U-PE | Devices with UNI: Add<br>All other Devices: Add | ☑ |

Usage notes:

- The templates/data files are selected by clicking on the Add link next to the desired option. The subsequent steps are the same as provided in Associating Templates and Data Files to a Policy, page B-5.

- This features is not applicable for device roles other than U-PE and PE-AGG. In Figure B-7 the N-PE role only displays a single Add link in the Template/Data File column.
- For backward compatibility, when editing or viewing old and existing policies, for U-PE and PE-AGG devices, associated templates/data files will display under the All other Devices option.
- When you copy an existing policy, you can copy associated templates/data files (if any) from the All other Devices or Devices with UNI options of the existing policy into the new policy. This is similar to normal ISC behavior.
- You can associate templates (without data files) for either the All other Devices or Devices with UNI options or both.
- Selective determination of templates is supported in all L2VPN and FlexUNI/EVC policy types and service requests. For MPLS VPN, only MPLS PE-CE and MPLS PE-NoCE policies and service requests are supported. For the MPLS VPN PE-CE policy type, this feature is applicable if the PE is or is not associated with an NPC.This feature is not available for Multi-VRFCE policies and service requests.

The following notes describe how this feature is supported in the service request workflow:

- During service request creation, selective templates are differentiated based on the devices having a UNI interface or having both UNI and NNI interfaces for the U-PE and PE-AGG device roles. Templates in the policy are copied to the respective devices functioning in the specified roles. There is no behavioral change for devices of other roles.
- The selective determination of templates is not applicable for service request modification scenarios, as after the service request is created, it is the user’s decision to make any changes for templates configured on devices.

# Using Templates with Service Requests

This section provides information on templates and data files with a service request. It contains the following sections:

- Overview of Template Use in Service Requests, page B-10
- Using Templates and Data Files with Service Requests, page B-14

## Overview of Template Use in Service Requests

This section provides overview information about template usage in service requests. It covers the following topics:

- Associating Templates to a Service Request, page B-11
- Associating Subtemplates During Service Provisioning, page B-11
- Creating Data Files During Service Request Creation, page B-12
- Using Negate Templates to Decommission Template Configurations, page B-13
- Using Templates and Data Files with Service Requests, page B-14

For details on how these features are implemented in the ISC GUI, see the section Using Templates and Data Files with Service Requests, page B-14.

## Associating Templates to a Service Request

The template mechanism in ISC provides a way to add additional configuration information to a device configuration generated by a service request. To use the template mechanism, the policy on which the service request is based must have been set to enable templates. Optionally, templates and data files to be used by the service request can be specified in the policy. During service request creation, templates/data files can be added to a device configuration if the operator has the appropriate RBAC permission to do so. See the section Choosing a Template in the Service Request Workflow, page B-14 for how to choose templates/data files in the service request workflow.

## Associating Subtemplates During Service Provisioning

All templates can be used by other templates as building blocks. The template using other templates is called a super template. The template being used is called a subtemplate. A new attribute in the Template Editor allows subtemplates to be associated with a super template. The super template instantiates all required subtemplates by passing values for the variables in the subtemplate. After instantiation, the super template puts the configlets generated for the subtemplate into the super template. ISC branches templates into subtemplates based on device type, line card type, port type, role type, and software versions. These optional attributes are set while creating the subtemplates. The subtemplates are selected based on the following matching criteria:

- Only exact matches are recognized for the card type and port type attributes. No wild card match is allowed for these attributes.
- Only an exact match is recognized for the device type attribute.
- For the software version attribute, the match is done for a software version equal to the current version, if available. If not, the previous highest version is matched.
- If exact matching attributes are not found, then the match proceeds with the criteria described in Table B-1. An information message listing the exactly matched subtemplates of the super-template is shown if and only if any of the matching criteria are met.
- If none of the attributes are matched, then the default subtemplate is applied.
- If no default subtemplate exists, a subtemplate with all null attribute values is matched.
- If none of the rows specified in the table match, then ISC looks for subtemplates that are marked as device default, or else version default. If no subtemplates are marked as such, then no matching subtemplates are picked. A warning message is displayed.

The matching criteria are summarized in Table B-1.

*Table B-1 Default SubTemplate Matching Criteria*

| Matching Order | Role Type | Device Type | Line Card | Port Type | Software Version |
|---|---|---|---|---|---|
| 1 | Exact Match | Exact Match | Exact Match | Exact Match | Exact Match |
| 2 | Exact Match | Exact Match | Exact Match | Exact Match | Previous Highest |
| 3 | Exact Match | Exact Match | Exact Match | No Values | Exact Match |
| 4 | Exact Match | Exact Match | Exact Match | No Values | Previous Highest |
| 5 | Exact Match | Exact Match | No Values | No Values | Exact Match |
| 6 | Exact Match | Exact Match | No Values | No Values | Previous Highest |
| 7 | Exact Match | Exact Match | No Values | No Values | No Values |
| 8 | Exact Match | No Values | Exact Match | Exact Match | Exact Match |
| 9 | Exact Match | No Values | Exact Match | Exact Match | Previous Highest |
| 10 | Exact Match | No Values | Exact Match | No Values | Exact Match |
| 11 | Exact Match | No Values | Exact Match | No Values | Previous Highest |
| 12 | Exact Match | No Values | No Values | No Values | Exact Match |
| 13 | Exact Match | No Values | No Values | No Values | Previous Highest |
| 14 | Exact Match | Default | No Values | No Values | No Values |
| 15 | Exact Match | No Values | No Values | No Values | Default |
| 16 | Exact Match | No Values | No Values | No Values | No Values |

Additional usage notes for subtemplates:

- ISC does not perform checks for the depth of subtemplates. Only one level of subtemplates is supported.
- No validations are done to check if the super template and subtemplate structures are cyclic.
- When the operator attempts to delete a subtemplate that is referenced by a super template, a warning message is generated.
- Subtemplates can be modified.
- Subtemplates can be attached to multiple super templates.
- In the current release, multiple data files are not supported for subtemplates. If multiple data files are found, the service request automatically chooses the first data file (from a list of available data files, sorted alphabetically).

## Creating Data Files During Service Request Creation

The operator can create data files “on demand” during service request creation. If template(s) are attached to a service policy, and no data file(s) exist for the template(s), a wizard prompts the operator to enter values for variables. If data file(s) are created on demand during service request creation, it is possible to modify any or all of the variables during modification or redeployment of the service request.

The service request workflow supports dynamic creation of data files as follows:

- If a template is marked as non-editable in the policy on which the service request is based, the operator cannot edit it during service request creation. However, the name of template and data files are still visible, even though they cannot be modified.
- If a template is marked as editable in the policy, then (assuming appropriate RBAC permission) the operator can change the template/data files during service request creation.

The following points apply if the template is editable:

- If a template is associated with a service policy, and at least one data file exists for the template, the operator can select the appropriate data file during service request creation.
- If only one data file exists for the template, it is automatically selected.
- During service request creation, the operator can enter values for template variables.
- Optionally, if no data file exists for the template, the operator can create a new data file during service request creation. When the Datafile Chooser window is opened from Template Association window, a Create Datafile button is provided, which allows the new data file to be created.
- The Create Datafile button is only displayed if the operator has the appropriate RBAC permissions to create a data file.
-

See the section Creating a Data File in the Service Request Workflow, page B-16 for how to set up a data file in the service request workflow.

## Using Negate Templates to Decommission Template Configurations

To remove a configuration created from a template/data file, a negate template must be applied to the existing service. ISC automatically applies the appropriate negate template during the decommission of the service request. For instructions on how to use the ISC Template Manager to create negate templates, see the *Cisco IP Solution Center Infrastructure Reference, 6.0*.

When a template is associated in a policy or service request, the negate template automatically gets associated. During decommission of the service, the negate template is used for deployment. When decommissioning a service request associated with a template/data file, the negate template is automatically picked up dynamically, by searching for a template name having the name of the original template followed by a suffix .Negate. This takes place at deployment time. Negate templates are dynamically instantiated based on the device attributes of the template to which it is associated.

**Note** Optional attributes (such as device type, line card type, port type, and software version) applied to a template automatically apply to the corresponding negate template. The optional attributes cannot be applied directly to negate templates.

When a service is decommissioned, the appropriate negate template is deployed. The data file for a negate template is selected during deployment as follows:

- If only one data file is associated with the negate template, the data file is automatically selected.
- If multiple data files are found, the data file with same name as template is selected.
- If multiple data files are found but no data file with the same name exists, the template is skipped.
- If no data file is found, the template is skipped.

The following points cover the behavior of templates in various modification scenarios:

- If you change the template associated with a service request, the negate template automatically changes to the negate template of the newly selected template. In this case, ISC executes the negate template of the previously associated template, as well as the newly associated template.
- When a template or negate template is modified, the service request does not roll back the configuration changes made earlier through the template.
- When a service request is modified, the template command is always deployed. (See the remaining bullet items for some additional clarifications.)
- When a service request is modified without changing template/data file information, the template commands are not redeployed. The only a modification that triggers a change in template/data file results is the negation of the old template and the addition of new template commands in the device configlet.
- When the ForceTemplateDeploy DCPL property is turned ON then, irrespective of templates being modified, if a service request is modified, templates are re-deployed. However, negate templates are not necessarily re-deployed. Negate templates are deployed only when a link/attachment circuit in the service request is deleted, which implicitly means removing templates associated with the link being deleted as well. When the ForceTemplateDeploy DCPL property is turned OFF, negate templates are instantiated under the following conditions:
  - Deleting or decommissioning a link/attachment circuit in a service request.
  - Modifying templates (for example, delete existing templates and adding new ones to a link, or deleting only existing ones).
  - Rehoming links/devices in a service request that has associated templates.
- When a device is changed in a service request, the negate template is deployed for the old device, and the template is deployed for the new device.
- When a link in a service request is removed and a new link is added, a negate template is deployed for the deleted link and a template is deployed for the added link.

# Using Templates and Data Files with Service Requests

This section describes tasks related to templates, data files, and negate templates that can be performed in the service request workflow. The following tasks are covered:

- Choosing a Template in the Service Request Workflow, page B-14
- Creating a Data File in the Service Request Workflow, page B-16
- Decommissioning Service Requests with Added Templates, page B-18
- Viewing Templates from the Service Requests Window, page B-18

## Choosing a Template in the Service Request Workflow

When creating a service request, the workflow involves selecting a policy on which to base the service request, setting interface and other attributes, and so on. The specific windows and attributes presented in the workflow depend on the type of service request, such as L2VPN, VPLS, MPLS, or FlexUNI/EVC.

To associate templates and data files in a service request, you must select a link in the appropriate window of the Service Request Editor window, usually by clicking the **Add** link for the device.

**Note** There is no choice of options to selectively determine templates for U-PE and PE-AGG devices during the service request workflow. Templates are automatically copied from the policy, based on the presence of a UNI interface on the devices functioning in U-PE and PE-AGG roles. See the section Selectively Determining Templates for U-PE and PE-AGG Device Roles, page B-9, for more information on this feature.

Perform the following steps to choose the template(s)/data file(s) for the device(s):

**Step 1** Click the **Add** link in Template/Datafile column for a device.

The Add/Remove Templates window appears.

**Step 2** Click the **Add** button.

The Template Datafile Chooser window appears.

**Step 3** Navigate to a template in the folder tree and select it.

The template is listed in the right side of the GUI, along with any data files that are associated with it.

At this point, you can either select an existing data file, or click the **Create Data File** button to create a data file dynamically in the workflow. The rest of the steps in this section cover the case of selecting an existing template and data file. For instructions on how to create a data file dynamically, see the section Creating a Data File in the Service Request Workflow, page B-16.

**Step 4** Check the check box of a data file to choose it.

**Step 5** Click the **Accept** button to confirm the choice.

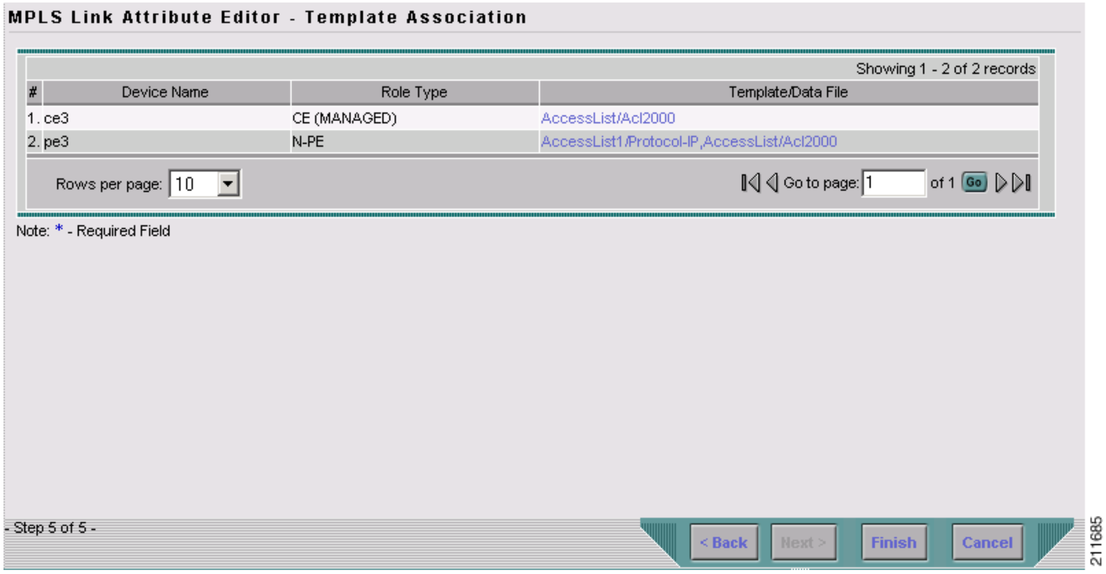
The template/data file combination appears in the Add/Remove Templates window.

**Step 6** To add additional templates/data files to the list, click the **Add** button and repeat the appropriate steps, as covered above.

**Step 7** When you are satisfied with selection of templates/data files, click the **OK** button in the Add/Remove Templates window.

The templates/data files appear in the Template/Datafile column of the Template Association window, as shown Figure B-8.

*Figure B-8 Sample Templates Association Window with Templates/Data Files Selected*



MPLS Link Attribute Editor - Template Association

Showing 1 - 2 of 2 records

| # | Device Name | Role Type | Template/Data File |
|---|---|---|---|
| 1. | ce3 | CE (MANAGED) | AccessList/Acl2000 |
| 2. | pe3 | N-PE | AccessList1/Protocol-IP,AccessList/Acl2000 |

Rows per page: 10 Go to page: 1 of 1

Note: * - Required Field

- Step 5 of 5 -

If multiple templates/data files are selected for a device, they appear as a comma-separated list, as shown in the figure.

**Step 8** Click the **Finish** button to create the service request with the template/data file selections you chose.

If the template associated to the service request is a super template comprising of one or more subtemplates, ISC displays a message confirming this.

For information about how templates/data files are instantiated when the service is deployed, see the information provided in the section Associating Templates to a Service Request, page B-11.

## Creating a Data File in the Service Request Workflow

During the final stage of setting the link attributes for a service request, the Template Association window appears, such as the one shown in Figure B-8. The Template Association window lists the devices comprising the link, the device roles, and the template(s)/data file(s) associated with the devices. You can choose the template(s)/data file(s) to be associated with the devices, as described in the section Choosing a Template in the Service Request Workflow, page B-14. If one of the templates selected in the Template Datafile Chooser window does not have an associated data file or if you would like create a new data file for it, you can do this dynamically in the workflow while setting up the service request.

Perform the following steps to dynamically set up a new data file for a template:

**Step 1** In the Template Association window, click the **Add** link in the Template/Datafile column for a device.

(If a template was previously selected for a device, click the link for the template name.)

The Add/Remove Templates window appears.

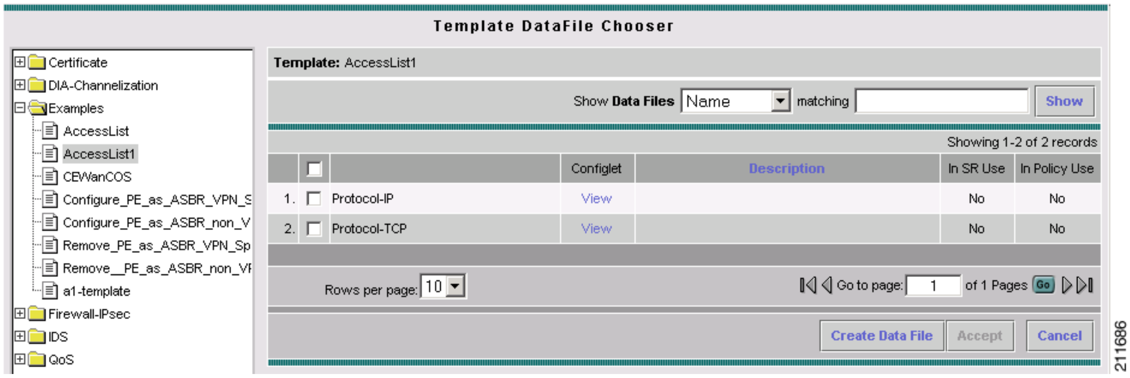
**Step 2** Click the **Add** button.

The Template Datafile Chooser window appears.

**Step 3** Navigate to a template in the folder tree and select it.

The template is listed in the right side of the GUI, along with any data files that are associated with it. This example uses the AccessList1 template in the Examples directory, as shown in Figure B-9.

*Figure B-9 Templates Datafile Chooser Window with No Data File Listed*



Template DataFile Chooser

- ⊞ Certificate
- ⊞ DIA-Channelization
- ⊟ Examples
  - AccessList
  - AccessList1
  - CEWanCOS
  - Configure_PE_as_ASBR_VPN_S
  - Configure_PE_as_ASBR_non_V
  - Remove_PE_as_ASBR_VPN_Sp
  - Remove__PE_as_ASBR_non_VI
  - a1-template
- ⊞ Firewall-IPsec
- ⊞ IDS
- ⊞ QoS

**Template:** AccessList1

Show **Data Files** Name ▾ matching [ ] Show

Showing 1-2 of 2 records

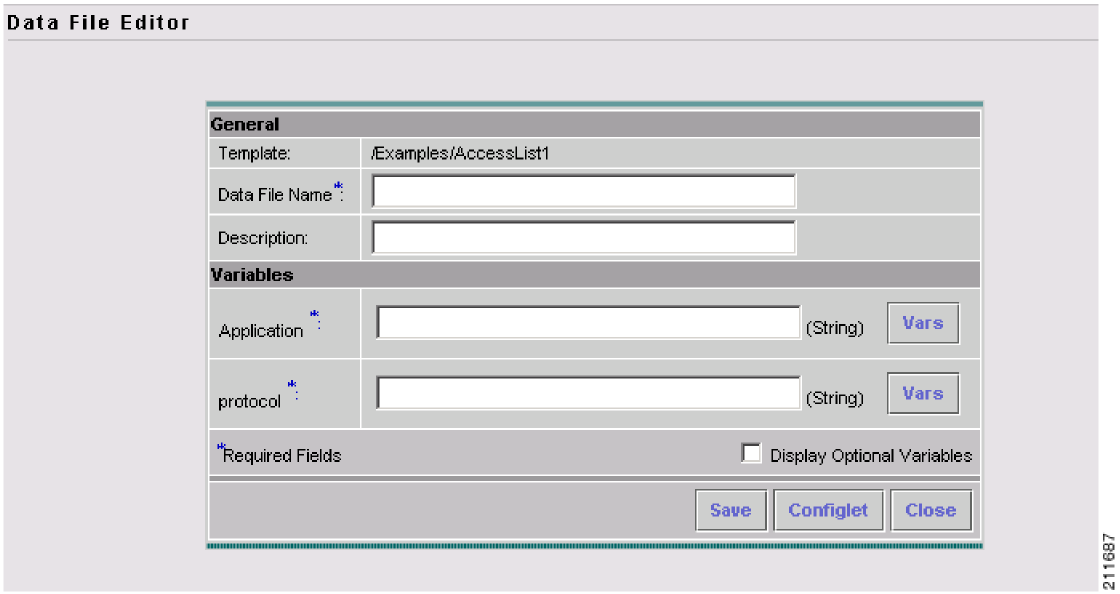
| | □ | | Configlet | Description | In SR Use | In Policy Use |
|---|---|---|---|---|---|---|
| 1. | □ | Protocol-IP | View | | No | No |
| 2. | □ | Protocol-TCP | View | | No | No |

Rows per page: 10 ▾ Go to page: 1 of 1 Pages Go

**Step 4** Click the **Create Data File** button to create a data file dynamically in the workflow.

The Data File Editor window appears, as shown in Figure B-10.

*Figure B-10 Data File Editor Window*



Data File Editor

| **General** | |
|---|---|
| Template: | /Examples/AccessList1 |
| Data File Name*: | |
| Description: | |
| **Variables** | |
| Application *: | (String) Vars |
| protocol *: | (String) Vars |

*Required Fields □ Display Optional Variables

**Step 5** At this point, you are in the standard workflow for creating a data file in ISC.

In the Date File Editor window, you can specify a name and description for the data file, set variable values, view the configlet, and so on. For details on how to perform these steps, see the chapter "Service Design" in the *Cisco IP Solution Center Infrastructure Reference, 6.0*.

**Step 6** When you have completed setting the attributes for the new data file, click **Save** and then **Close** to save this information and close the file; click **Configure** to show the configuration file; or click **Close** and then be sure to click **OK**, if you want to save the information you have created.

If you do not want to save this information, click **Close** and then click **Cancel**.

When the data file is saved, the Template Datafile Chooser window appears with the newly created data file listed.

## Decommissioning Service Requests with Added Templates

This section describes how to decommission ISC service requests that have added templates.

**Note** For general information on how templates are used in ISC, see Chapter 6, "Service Design" and Appendix D, "Template Usage" in the *Cisco IP Solution Center Infrastructure Reference, 6.0.*

As mentioned in the *Cisco IP Solution Center Infrastructure Reference, 6.0,* "Template commands are treated independently from those associated with a service creation. Consequently, template commands must be removed separately from the device(s) during a service decommission. To remove prior template commands, a separate template is needed during a decommission process. Decommissioning a service request does not automatically remove the original template commands. A separate negate template needs to be added to the decommission process and the original templates must be removed. The negate template must contain the necessary NO commands to successfully remove any unwanted IOS commands added by the original template."

The standard way to create a service request with a template added is as follows:

1. Define the service policy.
2. Build a template with a data file (and also a negate template and data file).
3. Create the service request with the template added. The steps to do this are covered in relevant chapters of this guide.
4. Deploy the service request to which the template was added.

To decommission a deployed service request, including associated templates, you must perform the following steps.

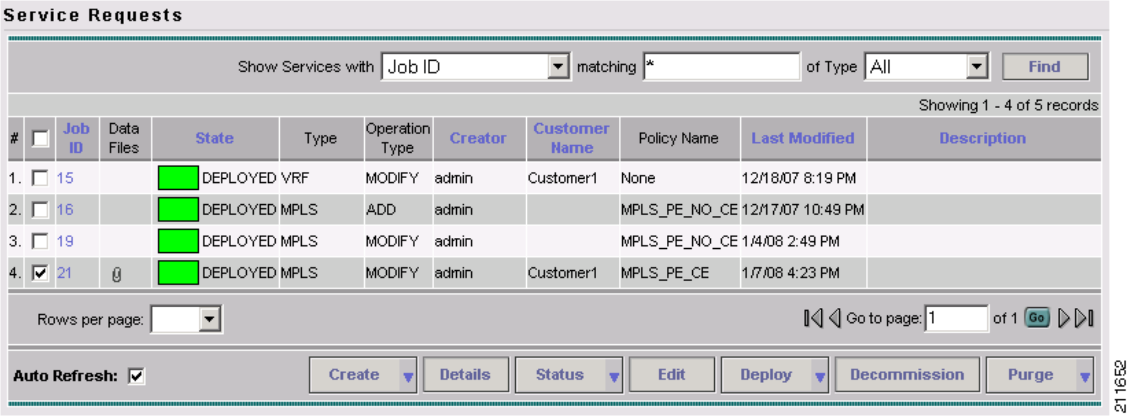
1. Create a negate template with data file (if one does not exist). This is used to remove the commands imposed by the original template. For an explanation of negate templates, see Chapter 4, "Using Templates" in the *Cisco IP Solution Center API Programmer Guide, 6.0.*
2. Decommission the service request. The negate template will be picked up dynamically.

   The service request remains in the **Requested** state, but changed to an Operation Type of Delete.
3. Deploy the service request. This decommissions the service request and downloads the negate template, which removes the original template commands.

## Viewing Templates from the Service Requests Window

In the Service Requests window, a paper clip icon appears in the Data Files column if a service request has one or more templates associated with it, as shown in Figure B-11.

*Figure B-11 Service Requests Window with Data Files Column*



Show Services with Job ID matching * of Type All

Showing 1 - 4 of 5 records

| # | ☐ | Job ID | Data Files | State | Type | Operation Type | Creator | Customer Name | Policy Name | Last Modified | Description |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1. | ☐ | 15 | | 🟩 DEPLOYED | VRF | MODIFY | admin | Customer1 | None | 12/18/07 8:19 PM | |
| 2. | ☐ | 16 | | 🟩 DEPLOYED | MPLS | ADD | admin | | MPLS_PE_NO_CE | 12/17/07 10:49 PM | |
| 3. | ☐ | 19 | | 🟩 DEPLOYED | MPLS | MODIFY | admin | | MPLS_PE_NO_CE | 1/4/08 2:49 PM | |
| 4. | ☑ | 21 | 📎 | 🟩 DEPLOYED | MPLS | MODIFY | admin | Customer1 | MPLS_PE_CE | 1/7/08 4:23 PM | |

Rows per page:

Go to page: 1 of 1 Go

Auto Refresh: ☑

**Note** You can use the **Show Services with** field to search for service requests that have a specific data or template file. Choose **Data File Name** or **Template Name** from the drop-down list and enter a search string in the **matching** field. The matching field is not case-sensitive and supports wildcards (*). You can further limit the search by using the **of Type** field to confine the search to a particular service type. When listing service requests using Template Name, provide the entire path of the template file location (for example: examples\template, where examples is the folder name and template implies the template name).

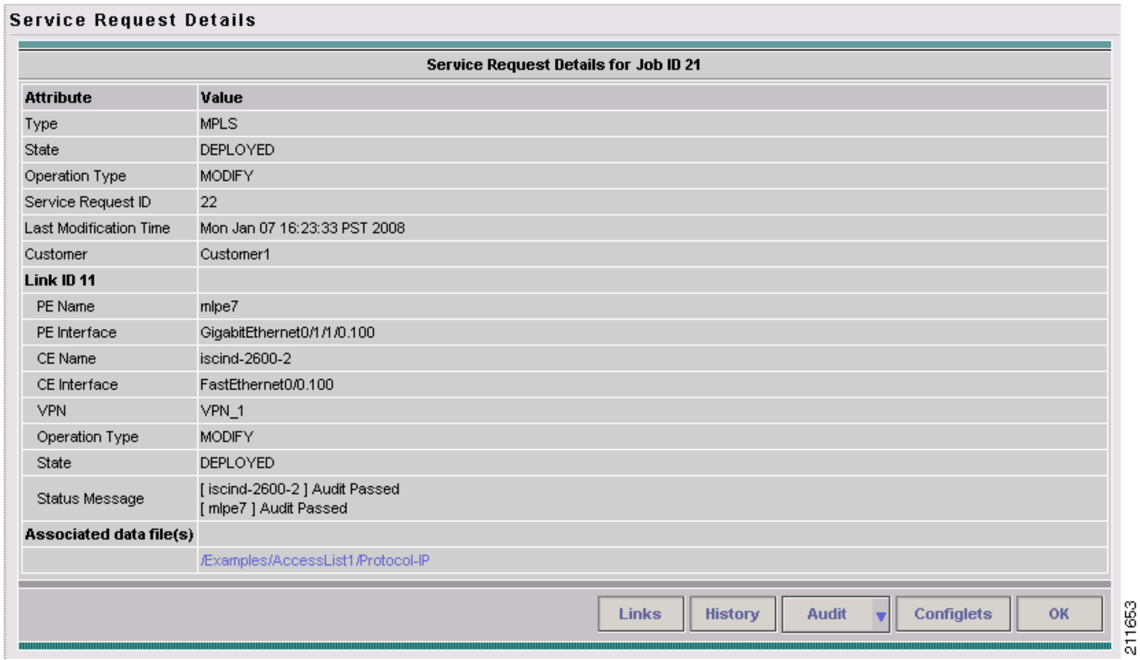
To view the configlet(s) for the template(s) associated with a service request, perform the following steps.

**Step 1** In the Service Requests window, check the check box for a service request with an associated template, as indicated by a paper clip icon in the Data Files column.

**Step 2** Click the **Details** button.

The Service Request Details window appears, as shown in Figure B-12.

*Figure B-12 Service Request Details Window*



Service Request Details

Service Request Details for Job ID 21

| Attribute | Value |
|---|---|
| Type | MPLS |
| State | DEPLOYED |
| Operation Type | MODIFY |
| Service Request ID | 22 |
| Last Modification Time | Mon Jan 07 16:23:33 PST 2008 |
| Customer | Customer1 |
| **Link ID 11** | |
| PE Name | mlpe7 |
| PE Interface | GigabitEthernet0/1/1/0.100 |
| CE Name | iscind-2600-2 |
| CE Interface | FastEthernet0/0.100 |
| VPN | VPN_1 |
| Operation Type | MODIFY |
| State | DEPLOYED |
| Status Message | [ iscind-2600-2 ] Audit Passed<br>[ mlpe7 ] Audit Passed |
| **Associated data file(s)** | |
| | /Examples/AccessList1/Protocol-IP |

The Associated data file(s) row displays a link for each data file associated with the service request, as shown in the figure.

**Step 3** Click a data file link to display the configlet for the template.

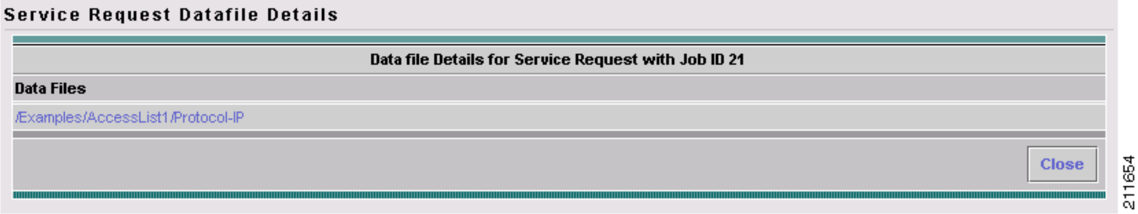
**Step 4** After viewing the configlet, click **OK** to close the configlet display window.

**Step 5** Click **OK** to close the Service Request Details window.

**Step 6** As an alternative, you can access the data files associated with a service request by clicking on the paper clip icon in the Service Requests window.

The Service Request Datafile Details window appears, as shown in Figure B-13.

*Figure B-13 Service Request Datafile Details Window*



Service Request Datafile Details

Data file Details for Service Request with Job ID 21

| Data Files |
|---|
| /Examples/AccessList1/Protocol-IP |

The window displays only a list of the data files associated with the service request.

**Step 7** Click a data file link to display the configlet for the template.

**Step 8** After viewing the configlet, click **OK** to close the configlet display window.

**Step 9** Click **Close** to close the Service Request Datafile Details window and return to the Service Requests window.

APPENDIX C

# Setting Up VLAN Translation

This appendix describes how to set up VLAN translation for L2VPN ERS (EVPL) services. It contains the following sections:

- VLAN Translation Overview, page C-1
- Setting Up VLAN Translation, page C-1
- Platform-Specific Usage Notes, page C-6

**Note** For helpful information to be aware of before you create policies and services using VLAN translation, review Platform-Specific Usage Notes, page C-6.

## VLAN Translation Overview

VLAN translation provides flexibility in managing VLANs and Metro Ethernet-related services. There are two types of VLAN translation—one is 1-to-1 translation (1:1), and the other one is 2-to-1 translation (2:1). This feature is available for L2VPN ERS (EVPL) (with and without a CE). The behavior of L2VPN ERS (EVPL) service remains the same, even though it is true that it is possible now for one Q-in-Q port to be shared by both EWS (EPL) and ERS (EVPL) service. VLAN translation is only for an Ethernet interface, not for other types of interfaces, such as ATM and Frame Relay.

With 1:1 VLAN translation, the VLAN of the incoming traffic (CE VLAN) is replaced by another VLAN (PE VLAN). It means the service provider is now able to handle the situation where incoming traffic from two different customers share the same CE VLAN. The SP can map these two CE VLANs to two different PE VLANs, and customer traffic will not be mixed.

With 2:1 VLAN translation, the double tagged (Q-in-Q) traffic at the U-PE UNI port can be mapped to different flows to achieve service multiplexing. The translation is based on the combination of the CE VLAN (inner tag) and the PE VLAN (outer tag). Without this translation, all the traffic from a Q-in-Q port can only go to one place because it is switched only by the outer tag.

## Setting Up VLAN Translation

The following sections described how to create and manage policies and service requests to support VLAN translation:

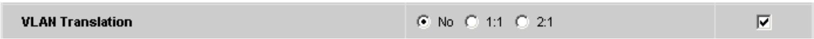
- Creating a Policy, page C-2
- Creating a Service Request, page C-3

- Modifying a Service Request, page C-5
- Deleting a Service Request, page C-5

# Creating a Policy

VLAN translation is specified during policy creation for L2VPN for ERS (EVPL) (with and without a CE). The L2VPN (Point to Point) Editor window contains a new option called **VLAN Translation**. (See Figure C-1.)

***Figure C-1 VLAN Translation Option in the L2VPN (Point to Point) Editor Window***



**VLAN Translation** ● No ○ 1:1 ○ 2:1 ☑

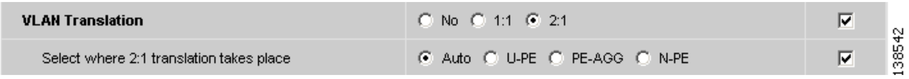
There are three options for VLAN translation:

- **No**—This is the default choice. No VLAN translation is performed.

**Note** If you choose **No** and you do not want to deal with any behavior related to VLAN translation during service request creation, then uncheck the **Editable** check box. This is the recommendation when you choose no VLAN translation.

- **1:1**—1:1 VLAN translation. The VLAN of the incoming traffic (CE VLAN) is replaced by another VLAN (PE VLAN). The specification of the VLAN translation is done during the creation of the service request for the policy, as covered in Creating a Service Request, page C-3.
- **2:1**—2:1 VLAN translation. The double tagged (Q-in-Q) traffic at the U-PE UNI port can be mapped to different flows to achieve service multiplexing. When you choose 2:1 VLAN translation, the L2VPN (Point to Point) Editor window dynamically changes to enable you to choose where the 2:1 VLAN translation takes place. (See Figure C-2.)

***Figure C-2 Choose Where 2:1 VLAN Translation Takes Place***



**VLAN Translation** ○ No ○ 1:1 ● 2:1 ☑

Select where 2:1 translation takes place ● Auto ○ U-PE ○ PE-AGG ○ N-PE ☑

The choices for where 2:1 VLAN translation takes place are:

- **Auto** (This is the default choice.)
- **U-PE**
- **PE-AGG**
- **N-PE**

If you choose **Auto**, the 2:1 VLAN translation takes place at the device closest to the UNI port. The other choices come into play only when there is more than one place that 2:1 VLAN translation can be done. If there is only one place where the translation can be done, the choice is ignored.

The actual VLAN values are specified when you create a service request based on this policy. See Creating a Service Request, page C-3.

# Creating a Service Request

When you create a service request based on an L2VPN ERS (EVPL) policy, the VLAN options can be changed if they were set to be editable in the policy. You can overwrite the policy information for the VLAN translation type and the place where translation occurs. This flexibility allows the following provisioning:

- One AC can have 2:1 VLAN translation, while the other AC can have no VLAN translation or 1:1 VLAN translation.
- The VLAN translation for one AC can be on the UNI box, while the translation for the other AC can be on the PE-AGG.

**Note** Note these modifications can happen only when a new service request is created. They are not allowed during the modification of an existing service request.

The specification of the VLAN translation happens during the creation of the service request within the Link Attributes window. At that point, you can specify which VLAN is translated to which VLAN. The Link Attributes window is accessed after the UNI port is selected on the Attachment Tunnel Editor window. Because you can set the VLAN translation type after the UNI selection, the UNI port display list does not exclude any type for the UNI port. This is because:

- The UNI port list has to include the regular trunk port, in case you later (on the Link Attributes window) decide to perform no VLAN translation or 1:1 VLAN translation.
- The UNI port list has to include an EWS (EPL) (Q-in-Q) port, in case you decide to do 2:1 VLAN translation.

Even though you have all the ports to start with for VLAN translation, you must choose specific types of ports, based on the type of VLAN translation. More specifically:

- For no VLAN translation and 1:1 VLAN translation, you must choose an empty port or a trunk port as the UNI.
- For 2:1 VLAN translation, you must choose an empty port or a Q-in-Q port as the UNI port.

To help determine the proper port to use, you can click the **Details** button on the Attachment Tunnel Editor window to display the port type and associated service with that port.

The following sections show how the VLAN translation is defined on the Link Attribute window for the different types of VLAN translation.

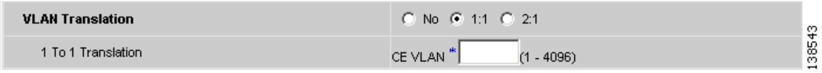
## No VLAN Translation

When you choose no VLAN translation, no additional information needs to be provided.

## 1:1 VLAN Translation

When you choose 1:1 VLAN translation, the window dynamically changes, as shown in Figure C-3.

*Figure C-3 CE VLAN to be Translated From*

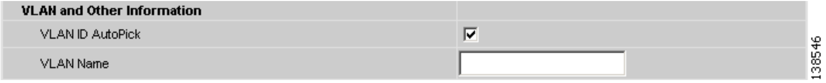


| | |
|---|---|
| **VLAN Translation** | ○ No ● 1:1 ○ 2:1 |
| 1 To 1 Translation | CE VLAN * [ ] (1 - 4096) |

In the empty field, you must enter which CE VLAN is to be translated from. The VLAN number must be a number from 1 to 4096.

The PE VLAN that the CE VLAN is to be translated to can be "auto picked" or manually entered. Check the **VLAN ID AutoPick** check box above (on the Link Attributes window) to have PE VLAN automatically assigned. (See Figure C-4.)

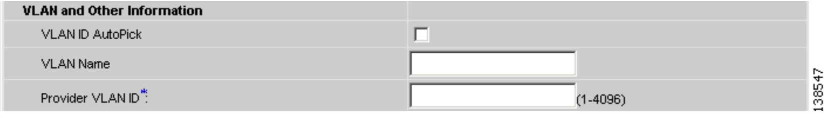
*Figure C-4 Automatic Selection of the PE VLAN*



| | |
|---|---|
| **VLAN and Other Information** | |
| VLAN ID AutoPick | ☑ |
| VLAN Name | |

If you uncheck the **VLAN ID AutoPick** check box (see Figure C-5), the window displays a Provider VLAN ID, where you can manually enter the PE VLAN.

*Figure C-5 Manual Selection of the PE VLAN*



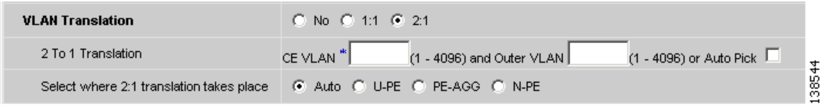
| | |
|---|---|
| **VLAN and Other Information** | |
| VLAN ID AutoPick | □ |
| VLAN Name | |
| Provider VLAN ID*: | (1-4096) |

Upon completion of the service request creation, ISC does an integrity check before saving the service request. For 1:1 VLAN translation, ISC rejects the service request if the CE VLAN has been used for another 1:1 VLAN translation on the same port.

## 2:1 VLAN Translation

When choosing 2:1 VLAN translation, the window dynamically changes, as shown in Figure C-6.

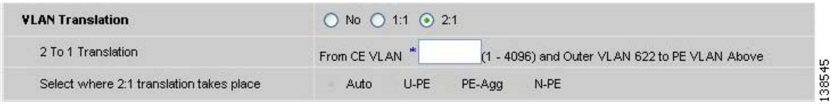
*Figure C-6 2:1 VLAN Translation Window*



| | |
|---|---|
| **VLAN Translation** | ○ No ○ 1:1 ● 2:1 |
| 2 To 1 Translation | CE VLAN * (1 - 4096) and Outer VLAN (1 - 4096) or Auto Pick □ |
| Select where 2:1 translation takes place | ● Auto ○ U-PE ○ PE-AGG ○ N-PE |

**Note** If the UNI port has been provisioned with EWS (EPL) service, the outer VLAN value is grayed out. (See Figure C-7.)

*Figure C-7 2:1 VLAN Translation with Outer VLAN Grayed Out*



| | |
|---|---|
| **VLAN Translation** | ○ No ○ 1:1 ● 2:1 |
| 2 To 1 Translation | From CE VLAN * (1 - 4096) and Outer VLAN 622 to PE VLAN Above |
| Select where 2:1 translation takes place | Auto U-PE PE-Agg N-PE |

In 2:1 VLAN translation, there are three VLANs involved:

- "A"—The CE VLAN to be translated from. You specify this in the "From CE VLAN field." For out-of-range translation, a value of "*" (asterisk character) should be provided

- "B"—The PE VLAN that is the outer VLAN of the Q-in-Q port. You specify this in the "Outer VLAN" field. You can choose this VLAN manually by entering a value, or you can choose the **AutoPick** check box to have one automatically assigned.
- "C"—The PE VLAN that the "A" and "B" VLANs are translated to. You specify this in the "VLAN and Other Information" section above (on the Link Attributes window). (See Figure C-4 and Figure C-5.)

You must specify VLAN "A" (the CE VLAN) and VLAN "C" (the PE VLAN translated to). For VLAN "B" (the Q-in-Q outer VLAN), what to specify depends on the UNI port type:

- If it is an empty port, you must specify VLAN "B."
- If it is an existing Q-in-Q port, then VLAN "B" has been defined, and it cannot be changed at this point.

Some additional comments on 2:1 VLAN translation:

- For 2:1 VLAN translation, if you build an ERS (EVPL) service on an empty port, then this UNI port will be provisioned as an ERS (EVPL) service. If you later add an EWS (EPL) service to the same port, the EWS (EPL) service will overwrite the previous ERS (EVPL) provisioning. The major difference between ERS (EVPL) and EWS (EPL) is the L2PT BPDU treatment. For ERS (EVPL), BPDU is blocked. For EWS (EPL), BPDU is tunneled.
- As an ERS (EVPL) service, the 2:1 VLAN translation can share the same port, just like a regular ERS (EVPL) port.
- An ERS (EVPL) 2:1 service can be added on top of an existing EWS (EPL) service.

Upon completion of the service request creation, ISC does an integrity check before saving the service request. For 2:1 VLAN translation, ISC rejects the service request if the CE VLAN and outer tag PE VLAN combination has been used for another 2:1 VLAN translation on the same port.

# Modifying a Service Request

For both 1:1 and 2:1 VLAN translation, you can perform the following modifications on an existing service request:

- Change to a new CE VLAN to be translated from.
- All other normal changes for a service request are permitted.

However, the following modifications are not allowed:

- You cannot change the VLAN translation type for a given AC. For instance, you cannot change from 2:1 to 1:1 VLAN translation.
- You cannot change the place where 2:1 VLAN translation occurs.

# Deleting a Service Request

During service request deletion, the following resources are released:

For 1:1 VLAN translation:

- The CE VLAN becomes available to be translated again.
- The PE VLAN is released.
- If the link being deleted is the last link on the UNI port, then this port is set to new.

For 2:1 VLAN translation:

- The CE VLAN becomes available to be translated again.
- The “translated to” PE VLAN is released.
- If the link being deleted is the last “CE-PE” pair on this UNI port, and there is no EWS (EPL) service on this port, then this port is set to new. In addition, the outer VLAN is released.

# Platform-Specific Usage Notes

VLAN translation is available on 7600 and 3750 ME platforms. The 7600 and 3750 ME have different ways to support VLAN translation. Not only is the command syntax different, but so is the place where the VLAN translation is carried out. On the 7600, for 1:1 VLAN translation, the operation is done on the PFC card. For 2:1 VLAN translation, the operation is done on the uplink GE-WAN (OSM module). On the 3750 ME, however, both translations occur on the uplinks (ES ports).

## VLAN Translation on the 3750

Be aware of the following points when performing VLAN translation on the 3750.

- The 3750 where VLAN translation occurs should be designated as a U-PE or PE-AGG role, not N-PE.
- VLAN translation on the up link (ES) port should be performed on the Gigabit 1/1/1 or Gigabit 1/1/2 port.
- If a 1:1 VLAN translation occurs on a ring that is made of 3750 PEs, all the 3750s use the ES port as uplink ports (the “east” and “west” ports) to connect other ring nodes.

## VLAN Translation on the 7600

Be aware of the following points when performing VLAN translation on the 7600.

- 1:1 VLAN translation always occurs on the UNI port. However, not every Ethernet interface will support 1:1 VLAN translation. Such support is dependent on the line card.
- 2:1 VLAN translation always occurs on the GE-WAN port. The port must be an NNI uplink port.
- 2:1 VLAN translation only occurs on a 7600 that is a U-PE or a PE-AGG, not an N-PE. The reason is when the 2:1 VLAN translation is performed on the GE-WAN interface, this interface can no longer perform L3VPN and L2VPN service using the translated new VLAN. The L3/L2VPN service has to be provisioned on another (N-PE) box.

## Failed Service Requests When Hardware Does Not Support VLAN Translation

For the 1:1 VLAN translation feature, a service request goes to the **Fail Deployed** state if the target hardware (line card) does not support the VLAN translation. The reason the service request goes to the **Fail Deployed** state instead of **Invalid** is that ISC does not know beforehand whether a particular line card will accept or reject the VLAN translation CLI commands. In this case, ISC attempts to push down the commands and the deployment fails. An **Invalid** status means ISC detects something wrong (in advance) and aborts the provisioning task. No CLI is pushed down in that case. This is a general behavior of ISC when a given hardware does not support a feature. In these cases, it is the user’s responsibility to select proper hardware to support the intended service.

APPENDIX D

# Terminating an Access Ring on Two N-PEs

This appendix describes how to terminate an access ring on two N-PEs for redundancy in case an access link goes down. It contains the following sections:

## Overview

ISC supports device-level redundancy in the service topology. This allows the service to remain active in case one access link should drop. This is accomplished through support for provisioning termination of access links against two different N-PEs. This is implemented by allowing an access ring to terminate on two different N-PEs. This may also be described as a "dual-homed access ring." The N-PEs are connected by a logical link using loopback interfaces on the N-PEs. The redundant link starts from a U-PE device and may, optionally, include PE-AGG devices. One attachment link is primary and one is secondary. The selection is made when the Named Physical Circuit (NPC) is created. The terminating device on the NPC acts as the primary N-PE, while the other N-PE on the same ring acts as the secondary N-PE.

For backward compatibility, ISC continues to support provisioning services without redundant links, as in previous releases.

**Note** N-PE redundancy is only supported for FlexUNI/EVC services.

**Note** MPLS service requests should not be deployed on access rings with two N-PEs. This might result in errors with the generated configlets.

Figure D-1 and Figure D-2 show two network topologies which illustrate redundancy, starting from a U-PE access node. Both topologies provide open segments for each uplink, starting from the U-PE and terminating on the N-PE devices. The N-PEs are logically connected via loopback interfaces. Services are configured on both of these Ethernet access links starting from the U-PE to two different N-PEs.

*Figure D-1 N-PE Redundancy, Starting at the U-PE*

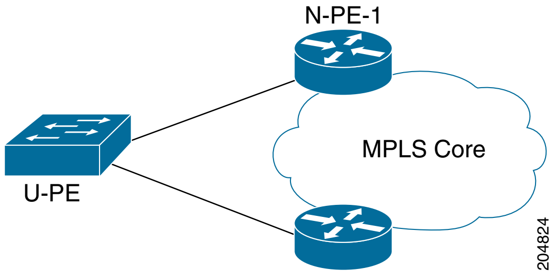
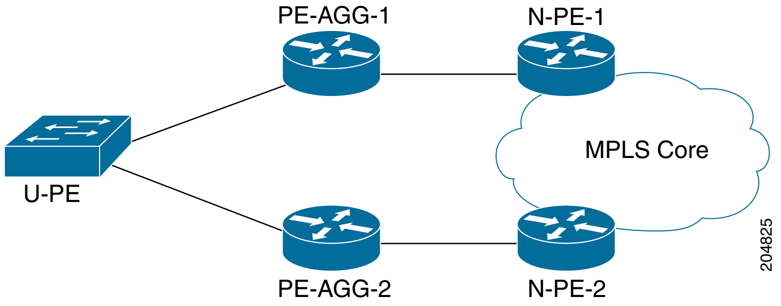


*Figure D-2 N-PE and PE-AGG Redundancy, Starting at the U-PE*



The first topology (N-PE redundancy starting at the U-PE, as shown in Figure D-1) provides the model of fault recovery for the N-PE device. As shown in the diagram, there are two different outgoing interfaces starting from the U-PE device. Each terminates at a different N-PE.

The second topology (N-PE and PE-AGG redundancy starting at the U-PE, as shown in Figure D-2) provides fault recovery for both the PE-AGG and N-PE devices. The service switches over from the primary to the secondary link when either the PE-AGG or the N-PE of the primary link fails.

For other network scenarios illustrating more complex topologies, see Additional Network Configurations and Sample Configlets, page D-5.

The following list provides additional details about the implementation:

- Using one U-PE and two N-PEs consumes one access link (AL).
- When creating a service on a U-PE, the user specifies an NPC to be used. If the topology includes an access ring with two N-PEs, then the service is configured on both N-PEs.
- For Ethernet over MPLS (EoMPLS) pseudowire (PW) services, if there is N-PE redundancy on both sides of the service provider network, two pseudowires are created. One N-PE is defined as primary and the other as secondary, in order to determine the how the pseudowires connect. If the user enables the PW Redundancy option, the primary and secondary on either end are also connected with pseudowire redundancy.
- For point-to-point (P2P) configurations, the two N-PEs use two separate pseudowires.
- ISC supports the case in which the service is configured identically (except for the access interface) on both N-PEs. This saves the user from having to enter data twice because the link attributes in the service request workflow are common for both N-PEs that are part of the attachment circuit.
- This feature is supported for both Cisco 7600 and Cisco ASR 9000 platforms. However, a single service cannot include both 7600 and ASR 9000 platforms.

- For the Cisco ASR 9000 platform, IOS XR version 3.7.3 and 3.9.0 are supported.

**Note** Check the on-line version of *Release Notes for Cisco IP Solution Center, 6.0,* for the most current information on device and platform support, in case updates have occurred since the publication of this guide.

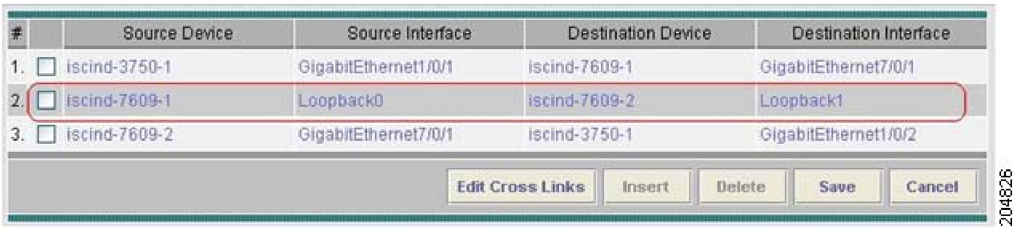
The implementation of this feature is covered in more detail in the following sections.

# Setting Up an NPC Access Ring with Two N-PEs

Terminating an NPC access ring on two N-PEs is achieved by using the standard method of setting up an NPC ring in ISC. The basic steps for doing this are described the *Cisco IP Solution Center Infrastructure Reference, 6.0.* Additional information is provided in this guide in the section Creating Named Physical Circuits, page 2-8.

In normal cases, a ring would be closed by connecting the devices via physical interfaces. When terminating an access ring on two different N-PEs, there is no need for a physical connection between the N-PEs. However, ISC requires that a virtual link must be created between the N-PEs, in order to close the ring. The virtual link is set up through the use of loopback interfaces. Figure D-3 shows the creation of an NPC ring with loopback interfaces.

***Figure D-3 NPC Ring with Loopback Interfaces Between N-PEs***



| # | | Source Device | Source Interface | Destination Device | Destination Interface |
|---|---|---|---|---|---|
| 1. | □ | iscind-3750-1 | GigabitEthernet1/0/1 | iscind-7609-1 | GigabitEthernet7/0/1 |
| 2. | □ | iscind-7609-1 | Loopback0 | iscind-7609-2 | Loopback1 |
| 3. | □ | iscind-7609-2 | GigabitEthernet7/0/1 | iscind-3750-1 | GigabitEthernet1/0/2 |

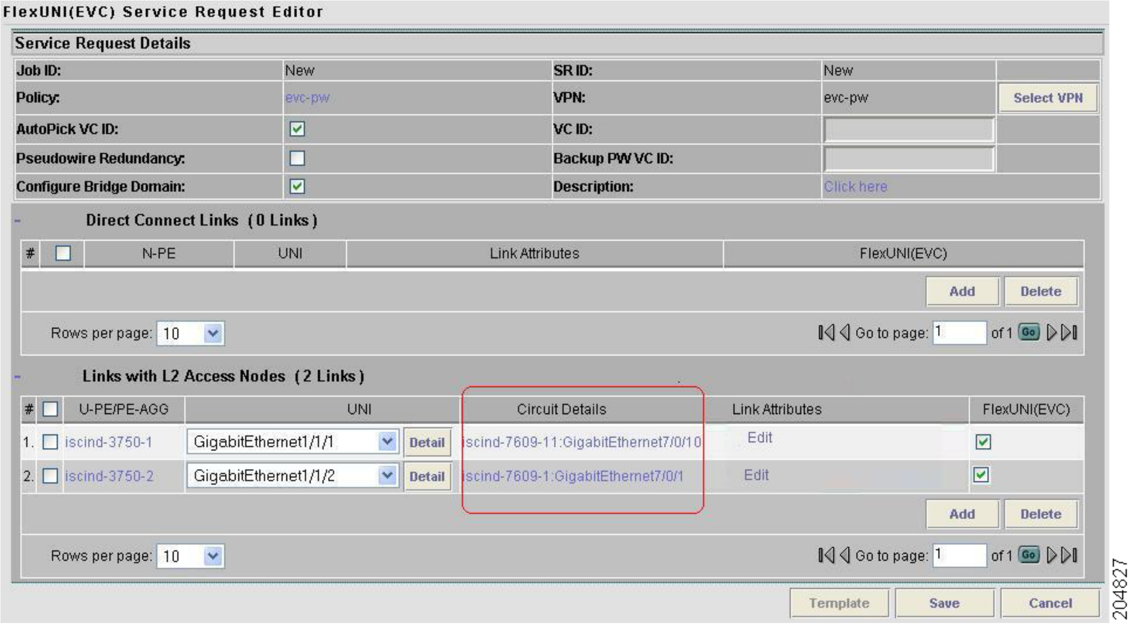
In order to use loopback interfaces in a ring in this manner, you must enable the DCPL property allowLoopbackIntfInNPC, which is accessed in the Host Configuration window under the folder /repository/mlshare. When this DCPL property is set to true, ISC allows the use of loopback interfaces in a ring.

**Note** Note that ISC does not generate any configlets onto the loopback interfaces during deployment of the service request.

# Using N-PE Redundancy in FlexUNI/EVC Service Requests

Using a dual-homed access ring in a FlexUNI/EVC service request does not require any change in the usual workflow in the ISC GUI. During creation of the FlexUNI/EVC service request, you select the NPC which is associated with an NPC access ring terminating on two N-PEs, as shown Figure D-4.

*Figure D-4 FlexUNI/EVC Service Request Editor Page, with N-PE Redundancy Option*



FlexUNI(EVC) Service Request Editor

Service Request Details

| | | | | |
|---|---|---|---|---|
| Job ID: | New | SR ID: | New | |
| Policy: | evc-pw | VPN: | evc-pw | Select VPN |
| AutoPick VC ID: | ☑ | VC ID: | | |
| Pseudowire Redundancy: | □ | Backup PW VC ID: | | |
| Configure Bridge Domain: | ☑ | Description: | Click here | |

Direct Connect Links (0 Links)

| # | □ | N-PE | UNI | Link Attributes | FlexUNI(EVC) |
|---|---|---|---|---|---|

Rows per page: 10 — Go to page: 1 of 1

Links with L2 Access Nodes (2 Links)

| # | □ | U-PE/PE-AGG | UNI | Circuit Details | Link Attributes | FlexUNI(EVC) |
|---|---|---|---|---|---|---|
| 1. | □ | iscind-3750-1 | GigabitEthernet1/1/1 Detail | iscind-7609-11:GigabitEthernet7/0/10 | Edit | ☑ |
| 2. | □ | iscind-3750-2 | GigabitEthernet1/1/2 Detail | iscind-7609-1:GigabitEthernet7/0/1 | Edit | ☑ |

Rows per page: 10 — Go to page: 1 of 1

Usage notes:

- The service is configured on both N-PEs of the access ring.
- Though there are two different N-PEs, only one access link is consumed.
- You can modify the configuration redundant N-PEs before or after deploying the service request. Modified configlets will be generated according to the changes made in service request.
- The destined N-PE device on the NPC used in the service request is treated as the primary N-PE. The other N-PE on the same ring is treated as the secondary N-PE. To change the primary and secondary N-PE, you must modify the attachment circuits in the service request.
- Configlets are generated according to the configuration specified in the service request. ISC generates identical configlets on both of the N-PEs in the attachment circuit (AC). The Link Attributes sections are common for both N-PEs.
- For FlexUNI/EVC services, N-PE redundancy is supported for PSEUDOWIRE and VPLS core connectivity types.
- In case of VPLS core connectivity, all N-PEs in NPC rings are configured to have Layer 2 Virtual Forwarding Interface (VFI), and all N-PEs on the same VPLS VPN participate in the VPLS service at the same time.
- In the case of PSEUDOWIRE core connectivity, the following notes apply:
  - If there is N-PE redundancy on both sides, a point-to-point pseudo wire (PW) will be configured between the N-PEs that were specified as the terminating N-PE devices during the NPC creation (between primary N-PEs). One more point-to-point PW will be configured between the N-PEs that were not specified as the terminating N-PE devices during NPC creation. The VC IDs of these pseudowires will be common.
  - If there is N-PE redundancy on only one side, then the Pseudowire Redundancy option must be checked in the GUI (in the Service Request Details section of the of the FlexUNI(EVC) Service Editor window). The primary PW will connect the primary N-PE of the dual-homed ring with

the N-PE of the single-homed ring, and the secondary PW will connect the secondary N-PE of the dual-homed ring with the N-PE of the single-homed ring. ISC will issue a warning message if you try to save the service request without enabling the Pseudowire Redundancy option.

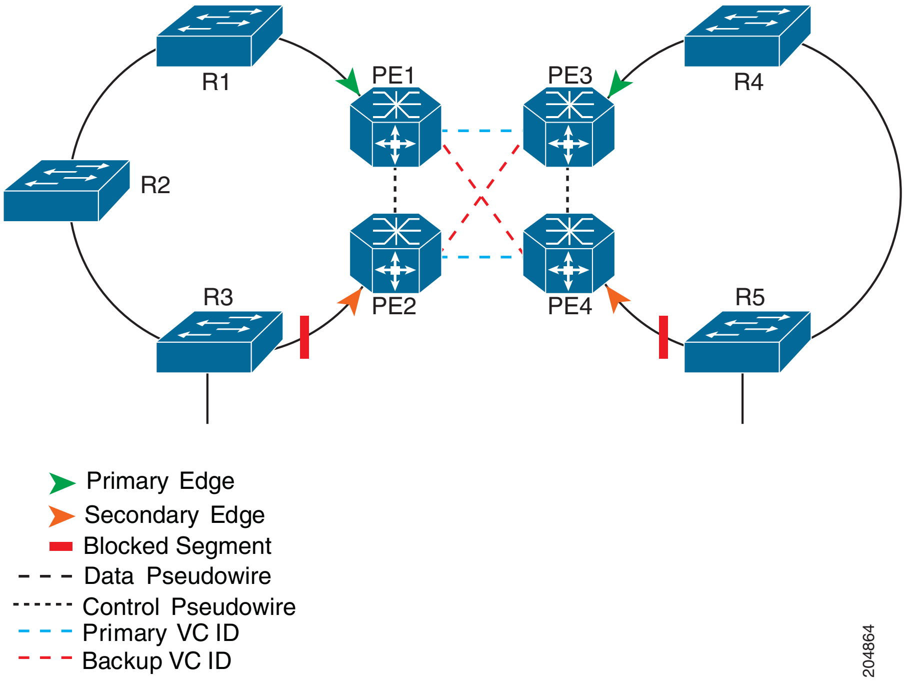
# Additional Network Configurations and Sample Configlets

This section provides additional network scenarios for reference, along with sample configlets for associated network devices.

## Example 1: Pseudowire Connectivity (A)

Figure D-5 illustrates a network configuration with pseudowire connectivity with dual-homed N-PEs on both sides of the network and with pseudowire redundancy.

*Figure D-5 Pseudowire Connectivity, Dual-Homed N-PEs on Both Sides of the Network, with Pseudowire Redundancy*



Sample configlets for the devices are provided below.

#### PE1

```
vlan <S-Vlan>
!
interface <UNI-to-R1>
    switchport
    switchport trunk encapsulation dot1q
    switchport mode trunk
    switchport trunk allowed vlan add <S-Vlan>
!
interface vlan <S-Vlan>
    xconnect <PE3 loopback> <PrimaryVcId> encapsulation mpls
    backup peer <PE4 loopback> <BackupVcId>
```

#### PE2

```
vlan <S-Vlan>
!
interface <UNI-to-R3>
    switchport
    switchport trunk encapsulation dot1q
    switchport mode trunk
    switchport trunk allowed vlan add <S-Vlan>
!
interface vlan <S-Vlan>
    xconnect <PE4 loopback> <PrimaryVcId> encapsulation mpls
    backup peer <PE3 loopback> <BackupVcId>
```

#### PE3

```
vlan <S-Vlan>
!
interface <UNI-to-R4>
    switchport
    switchport trunk encapsulation dot1q
    switchport mode trunk
    switchport trunk allowed vlan add <S-Vlan>
!
interface vlan <S-Vlan>
    xconnect <PE1 loopback> <PrimaryVcId> encapsulation mpls
    backup peer <PE2 loopback> <BackupVcId>
```

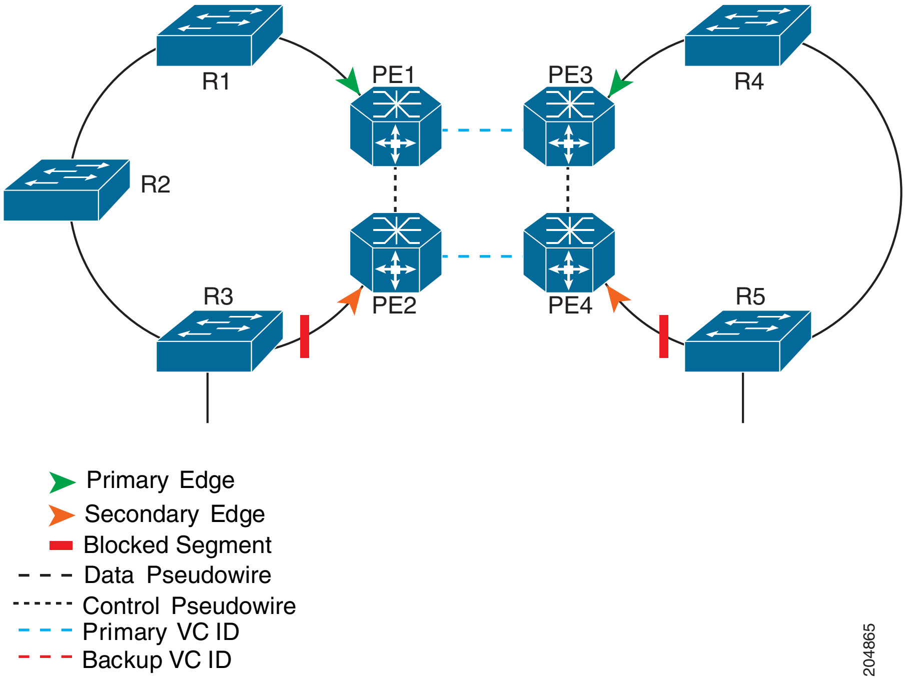
#### PE4

```
vlan <S-Vlan>
!
interface <UNI-to-R5>
    switchport
    switchport trunk encapsulation dot1q
    switchport mode trunk
    switchport trunk allowed vlan add <S-Vlan>
!
interface vlan <S-Vlan>
    xconnect <PE2 loopback> <PrimaryVcId> encapsulation mpls
    backup peer <PE1 loopback> <BackupVcId>
```

## Example 2: Pseudowire Connectivity (B)

Figure D-6 illustrates a network configuration using pseudowire connectivity, with dual-homed N-PEs on both sides of the network without pseudowire redundancy.

*Figure D-6 Pseudowire Connectivity, Dual-Homed N-PEs on Both Sides of the Network, with No Pseudowire Redundancy*



Sample configlets for the devices are provided below.

### PE1

```
vlan <S-Vlan>
!
interface <UNI-to-R1>
    switchport
    switchport trunk encapsulation dot1q
    switchport mode trunk
    switchport trunk allowed vlan add <S-Vlan>
!
interface vlan <S-Vlan>
    xconnect <PE3 loopback> <PrimaryVcId> encapsulation mpls
```

### PE2

```
vlan <S-Vlan>
!
interface <UNI-to-R3>
    switchport
    switchport trunk encapsulation dot1q
    switchport mode trunk
    switchport trunk allowed vlan add <S-Vlan>
!
interface vlan <S-Vlan>
    xconnect <PE4 loopback> <PrimaryVcId> encapsulation mpls
```

### PE3

```
vlan <S-Vlan>
!
interface <UNI-to-R4>
    switchport
    switchport trunk encapsulation dot1q
    switchport mode trunk
    switchport trunk allowed vlan add <S-Vlan>
!
interface vlan <S-Vlan>
    xconnect <PE1 loopback> <PrimaryVcId> encapsulation mpls
```

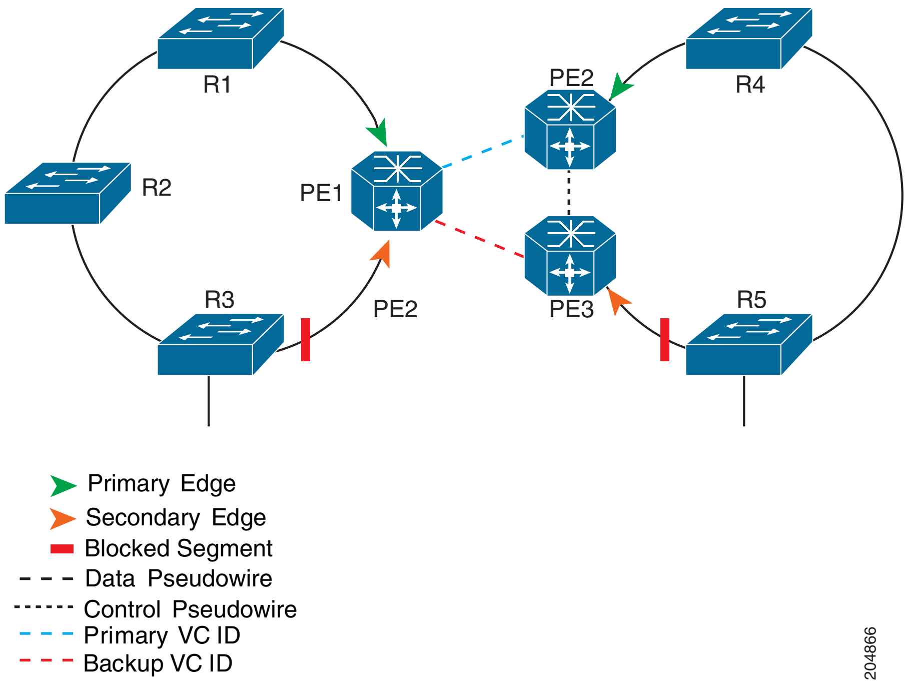
### PE4

```
vlan <S-Vlan>
!
interface <UNI-to-R5>
    switchport
    switchport trunk encapsulation dot1q
    switchport mode trunk
    switchport trunk allowed vlan add <S-Vlan>
!
interface vlan <S-Vlan>
    xconnect <PE2 loopback> <PrimaryVcId> encapsulation mpls
```

## Example 3: Pseudowire Connectivity (C)

Figure D-7 illustrates a network configuration using pseudowire connectivity with dual-homed N-PEs at one side of the network and with pseudowire redundancy.

*Figure D-7 Pseudowire Connectivity, Dual-Homed N-PEs on One Side of the Network, with Pseudowire Redundancy*



Sample configlets for the devices are provided below.

### PE1

```
vlan <S-Vlan>
!
interface <UNI-to-R1>
    switchport
    switchport trunk encapsulation dot1q
    switchport mode trunk
    switchport trunk allowed vlan add <S-Vlan>
!
interface vlan <S-Vlan>
    xconnect <PE2 loopback> <PrimaryVcId> encapsulation mpls
    backup peer <PE3 loopback> <BackupVcId>
```

### PE2

```
vlan <S-Vlan>
!
interface <UNI-to-R4>
    switchport
    switchport trunk encapsulation dot1q
    switchport mode trunk
    switchport trunk allowed vlan add <S-Vlan>
!
interface vlan <S-Vlan>
    xconnect <PE1 loopback> <PrimaryVcId> encapsulation mpls
```

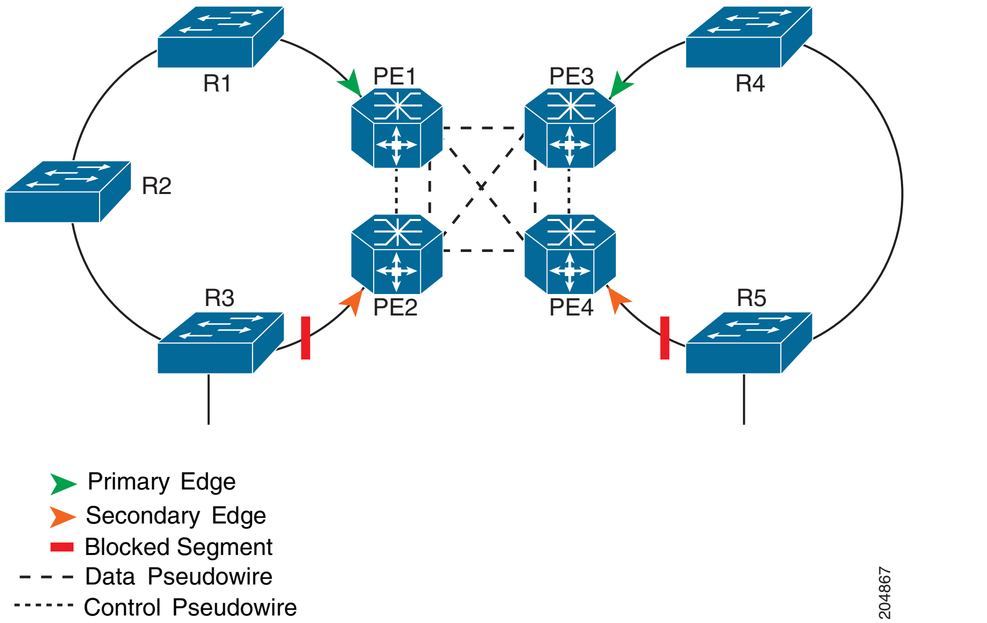
### PE3

```
vlan <S-Vlan>
!
interface <UNI-to-R5>
    switchport
    switchport trunk encapsulation dot1q
    switchport mode trunk
    switchport trunk allowed vlan add <S-Vlan>
!
interface vlan <S-Vlan>
    xconnect <PE1 loopback> <BackupVcId> encapsulation mpls
```

## Example 4: VPLS Connectivity

Figure D-8 illustrates a network configuration using VPLS connectivity with dual-homed N-PEs on both sides of the network.

*Figure D-8 VPLS Connectivity, Dual-Homed N-PEs on Both Sides of the Network*



Sample configlets for the devices are provided below.

### PE1

```
vlan <S-Vlan>
!
l2 vfi <VFI-ID> manual
    vpn id <S-Vlan>
    neighbor <PE2> encapsulation mpls
    neighbor <PE3> encapsulation mpls
    neighbor <PE4> encapsulation mpls
!
interface vlan <S-Vlan>
    xconnect vfi <VFI-ID>
!
interface <NNI-to-R1>
    switchport trunk allowed vlan add <S-Vlan>
```

### PE2

```
vlan <S-Vlan>
!
l2 vfi <VFI-ID> manual
    vpn id <S-Vlan>
    neighbor <PE1> encapsulation mpls
    neighbor <PE3> encapsulation mpls
    neighbor <PE4> encapsulation mpls
!
interface vlan <S-Vlan>
    xconnect vfi <VFI-ID>
!
interface <NNI-to-R3>
    switchport trunk allowed vlan add <S-Vlan>
```

### PE3

```
vlan <S-Vlan>
!
l2 vfi <VFI-ID> manual
    vpn id <S-Vlan>
    neighbor <PE1> encapsulation mpls
    neighbor <PE2> encapsulation mpls
    neighbor <PE4> encapsulation mpls
!
interface vlan <S-Vlan>
    xconnect vfi <VFI-ID>
!
interface <NNI-to-R5>
    switchport trunk allowed vlan add <S-Vlan>
```

### PE4

```
vlan <S-Vlan>
!
l2 vfi <VFI-ID> manual
    vpn id <S-Vlan>
    neighbor <PE1> encapsulation mpls
    neighbor <PE2> encapsulation mpls
    neighbor <PE3> encapsulation mpls
!
interface vlan <S-Vlan>
    xconnect vfi <VFI-ID>
!
interface <NNI-to-R4>
    switchport trunk allowed vlan add <S-Vlan>
```

APPENDIX E

# ISC Layer 2 VPN Concepts

This appendix provides an overview of ISC Layer 2 VPN concepts. It contains the following sections.

## Layer 2 Terminology Conventions

Layer 2 service provisioning for the IP Solution Center (ISC) consists of the Layer 2 Virtual Private Network (L2VPN) Service and the Virtual Private LAN Service (VPLS). The purpose of this section is to clarify the terminologies used in ISC, as well in the industry at large, for these services.

There are three sets of terminologies in use:

- The current Metro Ethernet Forum (MEF) terminology
- The former MEF terminology
- ISC terminology (which is close to the former MEF terminology)

### MEF Terminology Conventions

In general, for L2VPN services, the MEF supports two general Ethernet service type constructs:

- Ethernet Line (E-Line). Provides a point-to-point Ethernet Virtual Circuit (EVC).
- Ethernet LAN (E-LAN). Provides a multipoint-to-multipoint EVC.

Two Ethernet services are available for each type. These are distinguished by the means of service identification used at the user-to-network interface (UNIs), as follows:

- Port based. All-to-one bundling. These are referred to as “private.”
- VLAN-based. These services are multiplexed. The EVC is identified by a VLAN ID. These are referred as “virtual private.”

Table E-1 summarizes these relationships.

*Table E-1 Ethernet Service Definitions*

| Service Type | Port-Based | VLAN-Based |
|---|---|---|
| E-Line | Ethernet Private Line (EPL) | Ethernet Virtual Private Line (EVPL) |
| E-LAN | Ethernet Private LAN (EP-LAN) | Ethernet Virtual Private LAN (EVP-LAN) |

In addition to E-Line and E-LAN services, two additional service types are available for Layer 2:

- Frame Relay over MPLS (FRoMLS)
- ATM over MPLS (ATMoMPLS)

These service types are not covered in the current MEF documentation, in spite of the fact that the MEF has merged with the Frame Relay forum.

Formerly, another terminology was used by the MEF for Layer 2 services. Table E-3 maps the older terminology to the current one.

*Table E-2 MEF Ethernet Service Term Mappings*

| Current MEF Term | Former MEF Term |
|---|---|
| **L2VPN over MPLS Core** | |
| Ethernet Private Line (EPL) | Ethernet Wire Service (EWS) |
| Ethernet Virtual Private Line (EVPL) | Ethernet Relay Service (ERS) |
| ATM over MPLS (ATMoMPLS) | ATM over MPLS (ATMoMPLS) |
| Frame Relay over MPLS (FRoMPLS) | Frame Relay over MPLS (FRoMPLS) |
| **VPLS over MPLS Core** | |
| Ethernet Private LAN (EP-LAN) | Ethernet Wire Service (EWS) or<br>Ethernet Multipoint Service (EMS) |
| Ethernet Virtual Private LAN (EVP-LAN) | Ethernet Relay Service (ERS) or<br>Ethernet Relay Multipoint Service (ERMS) |
| **VPLS over Ethernet Core** | |
| Ethernet Private LAN (EP-LAN) | Ethernet Wire Service (EWS) |
| Ethernet Virtual Private LAN (EVP-LAN) | Ethernet Relay Service (ERS) |

For additional information about MEF conventions and additional useful background information on Metro Ethernet standards, see the MEF website at the following URL:

http://metroethernetforum.org

In particular, see the document *Metro Ethernet Services Definitions Phase 2* available under the section Information Center > MEF Technical Specifications on the MEF website for a useful presentation of Metro Ethernet terms and definitions.

## Mapping MEF Terminologies to Network Technologies

The MEF terminology only describes the outside characteristics of a service, that is, what the service looks like from the perspective of a customer looking in towards the user-to-network-interface (UNI) device. It does not describe how the service is implemented.

For details about how these service are implemented, see the following URL:

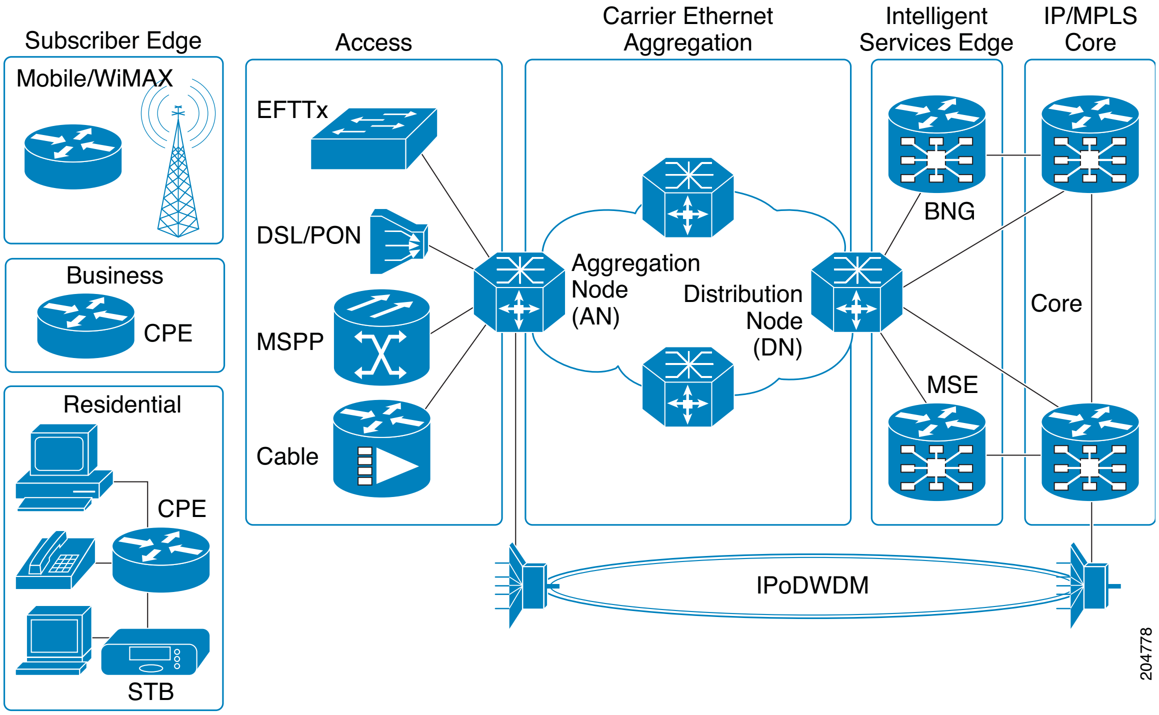
http://www.cisco.com/go/ce

In particular, see the documentation on that site on the subject of Cisco IP Next-Generation Network (NGN) Carrier Ethernet Design. The IP NGN Carrier Ethernet Design represents key elements of the Cisco IP NGN architecture that enable a best-in-class implementation for consistent service delivery optimized to meet the specific demands of each service. It is the end-to-end service transport foundation from the network access to the IP/MPLS core. This design provides integrated linkages with the service and application layer components to offer a converged, intelligent, reliable, and scalable network model to meet current and future network service requirements.

The IP NGN Carrier Ethernet Design (see Figure E-1) provides a platform-independent architecture and Ethernet-based services model across all Carrier Ethernet platforms. This allows service providers to optimize service transport with the intelligence of appropriate networking technologies (such as Ethernet, IP, MPLS, multicast, pseudowire, or hierarchical private virtual LAN services) to meet their business and quality-of-experience goals.

**Note** Real-world network implementations may implement only a subset of this very scalable architecture.

*Figure E-1 IP NGN Carrier Ethernet Design*



# ISC Terminology and Supported Network Types

This section discusses the ISC terminology for Layer 2 services and supported network types. ISC can provision the following service types:

- E-Line (EPL/EWS and EVPL/ERS)
- E-LAN (EP-LAN and EVP-LAN/ERMS)
- FRoMPLS
- ATMoMPLS

ISC also supports provisioning Ethernet services on a network that consists only of Ethernet switches (no MPLS), and this is referred to in ISC terminology as VPLS with L2 core.

**Note** For E-Line and E-LAN services, we recommend using the FlexUNI/EVC service policy type (see the appropriate chapters in this guide for how to create FlexUNI/EVC policies and service requests). You might have existing services that have been provisioned using the L2VPN and VPLS service policy types. These are still supported and can be maintained with those service types, but new services should use the FlexUNI /EVC service policy type. For ATM and FRoMPLS services, use the L2VPN service policy, as before.

In the ISC GUI and throughout this user guide, the naming conventions for these Ethernet services appear. These align closely with the earlier MEF conventions. This is expected to be updated in future releases of ISC. The equivalent terms used by the MEF forum are summarized in Table E-3, for reference.

*Table E-3 Ethernet Service Term Mappings*

| Term Used in ISC 5.2 GUI and This User Guide | Current MEF Equivalent Term |
|---|---|
| **L2VPN over MPLS Core** | |
| Ethernet Wire Service (EWS) | Ethernet Private Line (EPL) |
| Ethernet Relay Service (ERS) | Ethernet Virtual Private Line (EVPL) |
| ATM over MPLS (ATMoMPLS) | — |
| Frame Relay over MPLS (FRoMPLS) | — |
| **VPLS Over MPLS Core** | |
| Ethernet Wire Service (EWS) or<br>Ethernet Multipoint Service (EMS) | Ethernet Private LAN (EP-LAN) |
| Ethernet Relay Service (ERS) or<br>Ethernet Relay Multipoint Service (ERMS) | Ethernet Virtual Private LAN (EVP-LAN) |
| **VPLS over Ethernet Core** | |
| Ethernet Wire Service (EWS) | Ethernet Private LAN (EP-LAN) |
| Ethernet Relay Service (ERS) | Ethernet Virtual Private LAN (EVP-LAN) |

# L2VPN Service Provisioning

This section provides and overview of ISC provisioning for L2VPN services that provide Layer 2 point-to-point connectivity over an MPLS core. Cisco's Any Transport over MPLS (AToM) enables supports these services. These implementations, in turn, support service types, as follows:

- Ethernet Wire Service (EWS). The MEF term for this service is EPL.
- Ethernet Relay Service (ERS). The MEF term for this service is EVPL.
- ATM over MPLS (ATMoMPLS)
- Frame Relay over MPLS (FRoMPLS)

Instructions on creating policies and service requests for these services are provided in other chapters of the guide. For more information, see the following sections:

- Point-to-Point Ethernet (EWS and ERS) (EPL and EVPL), page E-5
- ATM over MPLS (ATMoMPLS), page E-8
- Frame Relay over MPLS (FRoMPLS), page E-9

## Point-to-Point Ethernet (EWS and ERS) (EPL and EVPL)

The EWS and ERS services (also known as EPL and EVPL, respectively, in MEF terminology) are delivered with the Cisco Metro Ethernet offering. The same network architecture can simultaneously provide both ERS (EPL) and EWS (EVPL) connections to diverse customers. Additionally, this Metro Ethernet infrastructure can be used for access to higher-level services, such as IP-based virtual private networking, public internet communications, Voice over IP, or a combination of all applications.

### Ethernet Wire Service (EWS or EPL)

An Ethernet Virtual Circuit (EVC) connects two physical User-to-Network Interfaces (UNI) such that the connection appears like a virtual private line to the customer. VLAN transparency and control protocol tunnelling are supplied by the implementation of 802.1Q-in-Q tag-stacking technology. Packets received on one UNI are transported directly to the other corresponding UNI.

The MEF term for this service is EPL.

### Ethernet Relay Service (ERS or EVPL)

An Ethernet Virtual Circuit (EVC) is used to logically connect endpoints, but multiple EVCs could exist per single UNI. Each EVC is distinguished by 802.1q VLAN tag identification. The ERS network acts as if the Ethernet frames have crossed a switched network, and certain control traffic is not carried between ends of the EVC. ERS is analogous to Frame Relay where the CE-VLAN tag plays the role of a Data-Link Connection Identifier (DLCI).

The MEF term for this service is EVPL.

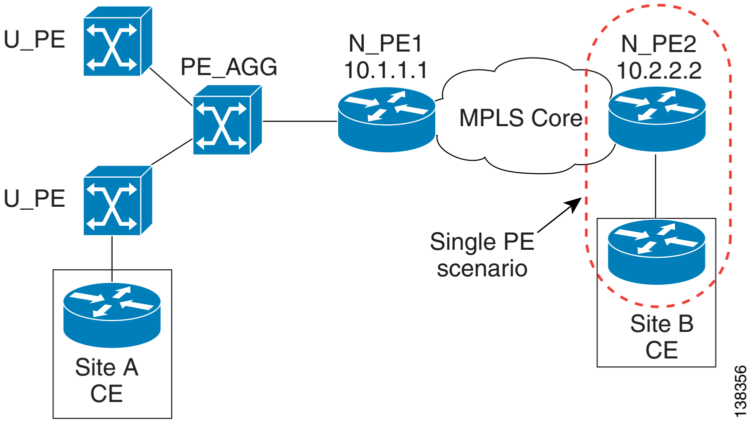
### Topology for L2VPN Ethernet Over MPLS (ERS and EWS) (EPL and (EVPL)

Ethernet Over MPLS (EoMPLS) is a tunnelling mechanism that allows the service provider to tunnel customer Layer 2 traffic though a Layer 3 MPLS network. It is important to remember that EoMPLS is a point-to-point solution only.

The following figures provide a reference for how EoMPLS is utilized. Ethernet Services can be distributed to the end customer in two ways.

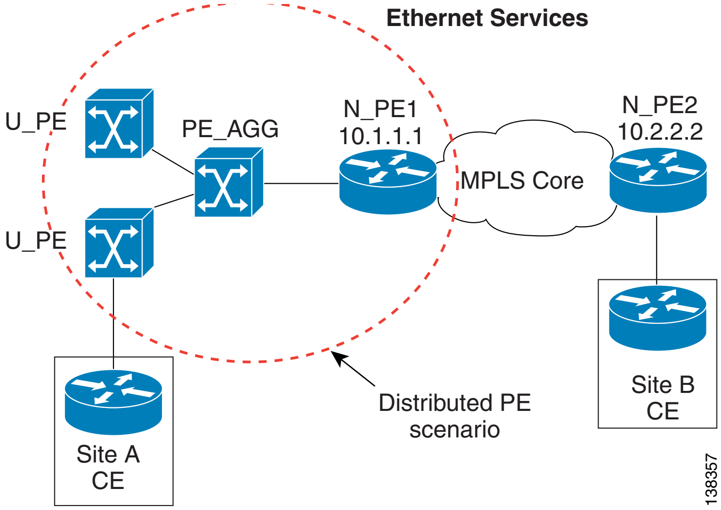
- Single PE scenario—The customer is directly connected to an Ethernet port on the N-PE in Figure E-2.

***Figure E-2 Single PE scenario***



- Distributed PE scenario—The end customer is connected through an Access Domain to the N-PE in Figure E-3. That is, there is a Layer 2 switching environment in the middle of CE and N-PE.

***Figure E-3 Distributed PE Scenario***



In both cases, a VLAN is assigned in one of the following ways:

- Automatically assigned by ISC from the VLAN pool that is predefined by the user.
- Manually assigned by the user through the GUI or the North Bound Interface (NBI).

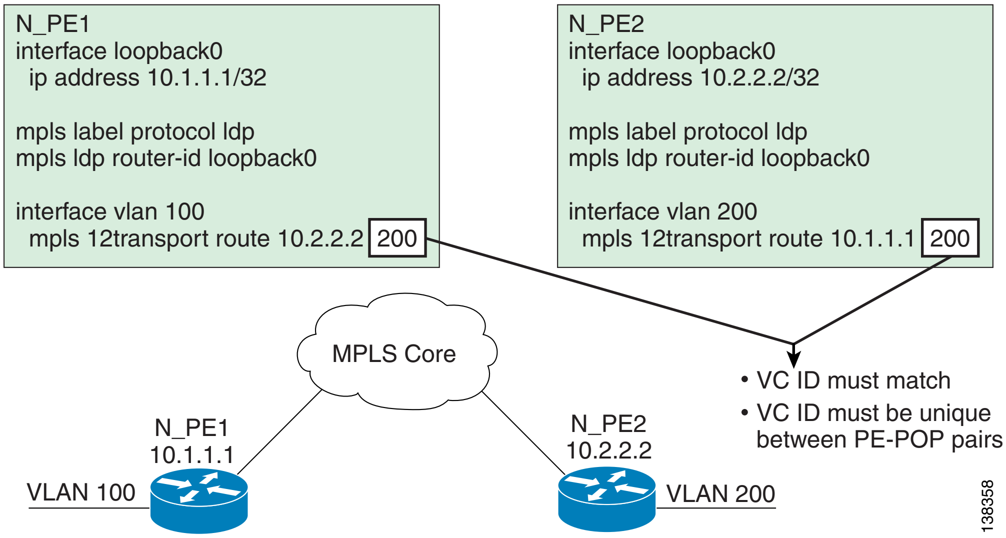
In EoMPLS, ISC creates a point-to-point tunnel and then targets the EoMPLS tunnel to the peer N-PE router through which the remote site can be reached. The remote N-PE is identified by its loopback address. In Figure E-4, N-PE1 and N-PE2 have 10.1.1.1 and 10.2.2.2 as loopback addresses. In Figure E-4, Site A has been allocated a VLAN-100 and Site B a VLAN-200. You can have different VLAN IDs at either end of the circuit because the VLANs have local significance only (that is, within the Ethernet access domain which is delimited by the N-PE).

For the N-PE that is serving Site A, a VLAN interface (Layer 3 interface) is created to terminate all L2 traffic for the customer, and an EoMPLS tunnel is configured on this interface.

**Note** This configuration is based on the Cisco 7600 Optical Services Router. Other routers, such as the Cisco 7200, have different configurations.

The VC ID that defines the EoMPLS tunnel is 200. (See Figure E-4.)

***Figure E-4 Ethernet over MPLS Configuration***



```
N_PE1
interface loopback0
  ip address 10.1.1.1/32

mpls label protocol ldp
mpls ldp router-id loopback0

interface vlan 100
  mpls 12transport route 10.2.2.2 200
```

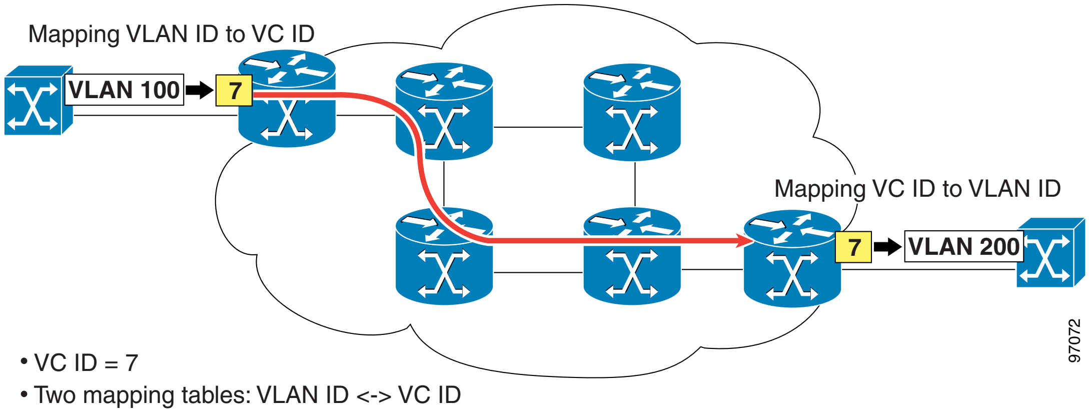
```
N_PE2
interface loopback0
  ip address 10.2.2.2/32

mpls label protocol ldp
mpls ldp router-id loopback0

interface vlan 200
  mpls 12transport route 10.1.1.1 200
```

Note that the VC ID has to be the same on both ends of the EoMPLS tunnel. On each N-PE, there is mapping done between the VLANs to the EoMPLS tunnel. (See Figure E-5.)

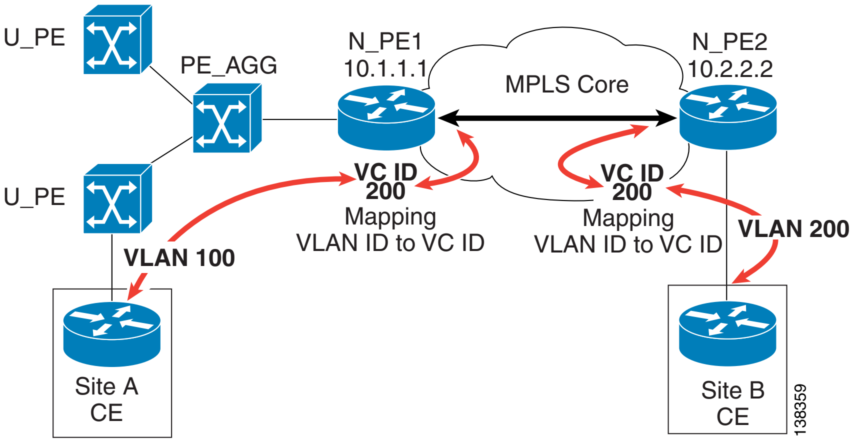
***Figure E-5 EoMPLS Tunnel***



For the overall connection, this mapping is: VLAN ID <-> VC ID <-> VLAN ID.

This VLAN-VC ID mapping lets the service provider reuse VLAN IDs in Access Domains. (See Figure E-6.)

*Figure E-6 VLAN-VC ID Mapping*



The VLAN IDs allocated and used at each access domain do not have to be the same.

## ATM over MPLS (ATMoMPLS)

With Cisco ATM over MPLS (ATMoMPLS), Cisco supports ATM Adaptation Layer 5 (AAL5) transport and Cell Relay over MPLS.

### AAL5

AAL5 allows you to transport AAL5 PDUs from various customers over an MPLS backbone. ATM AAL5 extends the usability of the MPLS backbone by enabling it to offer Layer 2 services in addition to already existing Layer 3 services. You can enable the MPLS backbone network to accept AAL5 PDUs by configuring the provider edge (PE) routers at both ends of the MPLS backbone.

To transport AAL5 PDUs over MPLS, a virtual circuit is set up from the ingress PE router to the egress PE router. This virtual circuit transports the AAL5 PDUs from one PE router to the other. Each AAL5 PDU is transported as a single packet.

### Cell Relay over MPLS

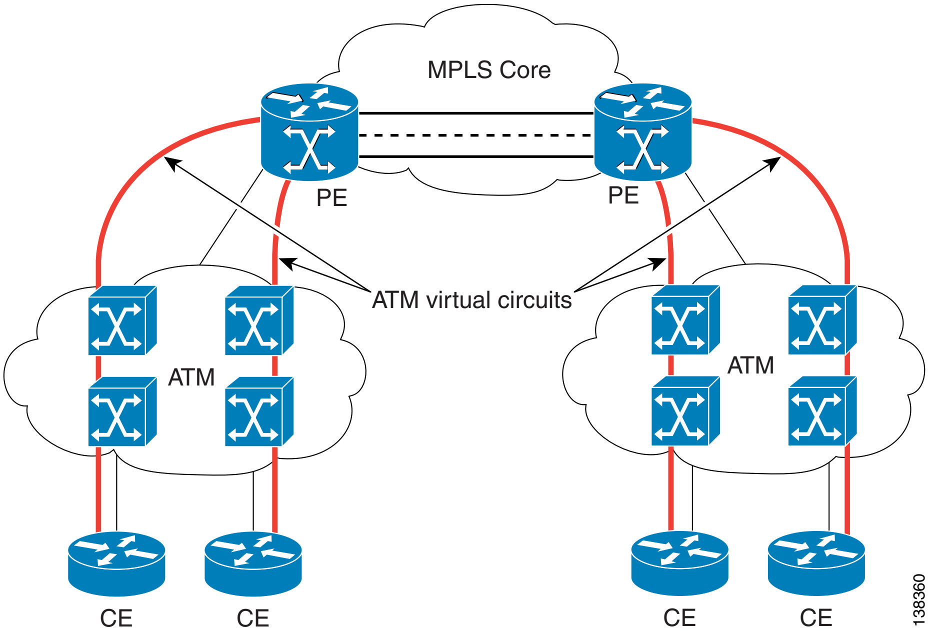
Cell Relay over MPLS allows you to transport ATM cells from various customers over an MPLS backbone. ATM Cell Relay extends the usability of the MPLS backbone by enabling it to offer Layer 2 services in addition to already existing Layer 3 services. You can enable the MPLS backbone network to accept ATM cells by configuring the provider edge (PE) routers at both ends of the MPLS backbone.

To transport ATM cells over MPLS, a virtual circuit is set up from the ingress PE router to the egress PE router. This virtual circuit transports the ATM cells from one PE router to the other. Each MPLS packet can contain one or more ATM cells. The encapsulation type is AAL0.

### Topology for ATMoMPLS

Only the single PE scenario is supported. (See Figure E-7.)

*Figure E-7 Configuring AAL5 and Cell Relay over MPLS*



## Frame Relay over MPLS (FRoMPLS)

With Cisco AToM for Frame Relay, customer Frame Relay traffic can be encapsulated in MPLS packets and forwarded to destinations required by the customer. Cisco AToM allows service providers to quickly add new sites with less effort than typical Frame Relay provisioning.

Frame Relay over MPLS enables a service provider to transport Frame Relay frames across an MPLS backbone. This extends the reachability of Frame Relay and allows service providers to aggregate frame transport across a common packet backbone. The service provider can integrate an existing Frame Relay environment with the packet backbone to improve operational efficiency and to implement the high-speed packet interfaces to scale the Frame Relay implementations.

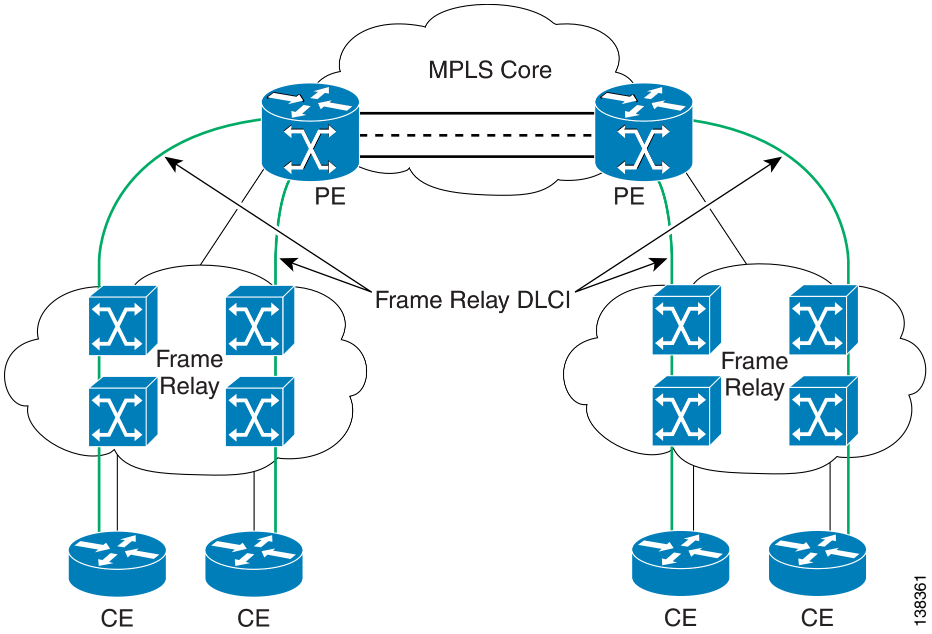
Transporting Frame Relay frames across MPLS networks provides a number of benefits, including:

- Frame Relay extended service.
- Aggregation to a higher speed backbone, such as OC-192, to scale Frame Relay implementations.
- Improved operational efficiency—the MPLS backbone becomes the single network that integrates the various existing networks and services.

### Topology for FRoMPLS

Only the single PE scenario is supported. (See Figure E-8.)

*Figure E-8 Frame Relay over MPLS*



# VPLS Service Provisioning

VPLS services are multipoint. They provide multipoint connectivity over an MPLS or an Ethernet core. These implementations, in turn, support service types, as follows:

- VPLS over MPLS core:
  - Ethernet Wire Service (EWS). This is also sometimes referred to as EMS, or Ethernet Multipoint Service. The MEF term for this service is EP-LAN.
  - Ethernet Relay Service (ERS). This is also sometimes referred to ERMS, or Ethernet Relay Multipoint Service. The MEF term for this service is EVP-LAN.
- VPLS over Ethernet core:
  - Ethernet Wire Service (EWS). The MEF term for this service is EP-LAN.
  - Ethernet Relay Service (ERS). The MEF term for this service is EVP-LAN.

Instructions on creating policies and service requests for these services are provided in other chapters of the guide.

VPLS is a multipoint Layer 2 VPN that connects two or more customer devices using EoMPLS bridging techniques. VPLS EoMPLS is an MPLS-based provider core, that is, the PE routers have to cooperate to forward customer Ethernet traffic for a given VPLS instance in the core. A VPLS essentially emulates an Ethernet switch from a user's perspective. All connections are peers within the VPLS and have direct communications. The architecture is actually that of a distributed switch. Multiple attachment circuits have to be joined together by the provider core. The provider core has to simulate a virtual bridge that connects these multiple attachment circuits together. To achieve this, all PE routers participating in a VPLS instance form emulated VCs among them.

A Virtual Forwarding Instance (VFI) is created on the PE router for each VPLS instance. PE routers make packet-forwarding decisions by looking up the VFI of a particular VPLS instance. The VFI acts like a virtual bridge for a given VPLS instance. More than one attachment circuit belonging to a given

VPLS can be connected to this VFI. The PE router establishes emulated VCs to all the other PE routers in that VPLS instance and attaches these emulated VCs to the VFI. Packet forwarding decisions are based on the data structures maintained in the VFI. All the PE routers in the VPLS domain use the same VC-ID for establishing the emulated VCs. This VC-ID is also called the VPN-ID in the context of the VPLS VPN.

For more information, see the following sections:

- Multipoint EWS (EP-LAN) for an MPLS-Based Provider Core, page E-11
- Multipoint ERS (EVP-LAN) for an MPLS-Based Provider Core, page E-11
- Topology for MPLS-Based VPLS, page E-11

## Multipoint EWS (EP-LAN) for an MPLS-Based Provider Core

With multipoint EWS (also known as EP-LAN in MEF terminology), the PE router forwards all Ethernet packets received from an attachment circuit, including tagged, untagged, and Bridge Protocol Data Unit (BPDU) to either:

- Another attachment circuit or an emulated VC if the destination MAC address is found in the L2 forwarding table (VFI).
- All other attachment circuits and emulated VCs belonging to the same VPLS instance if the destination MAC address is a multicast/broadcast address or not found in the L2 forwarding table.

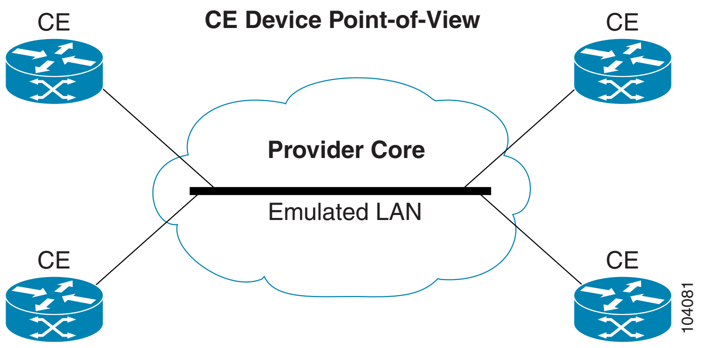
## Multipoint ERS (EVP-LAN) for an MPLS-Based Provider Core

With multipoint ERS (also known as EVP-LAN in MEF terminology), the PE router forwards all Ethernet packets with a particular VLAN tag received from an attachment circuit, excluding BPDU, to another attachment circuit or an emulated VC if the destination MAC address is found in the L2 forwarding table (VFI). If the destination MAC address is not found or if it is a broadcast/multicast packet, then it is sent on all other attachment circuits and emulated VCs belonging to the VPLS instance. The demultiplexing VLAN tag used to identify a VPLS domain is removed prior to forwarding the packet to the outgoing Ethernet interfaces or emulated VCs because it only has local significance.

## Topology for MPLS-Based VPLS

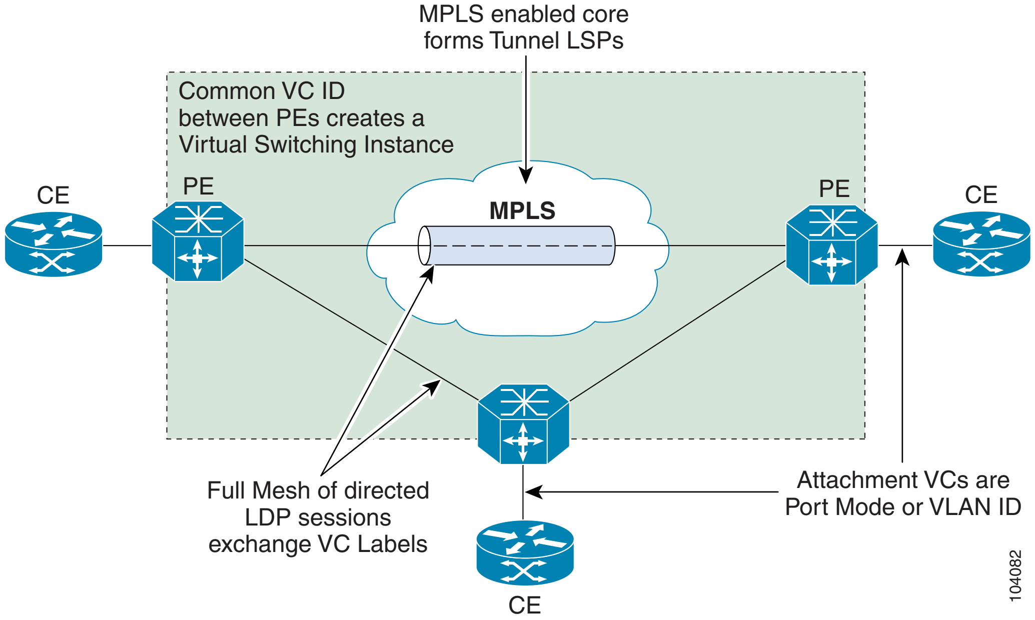
From a customer point of view there is no topology for VPLS. All the CE devices are connected to a logical bridge emulated by the provider core. Therefore, the CE devices see a single emulated LAN. (See Figure E-9.)

*Figure E-9* ***MPLS-Based VPLS Topology***



The PE routers must create a full-mesh of emulated virtual circuits (VCs) to simulate the emulated LAN seen by the CE devices. Forming a full-mesh of emulated VCs simplifies the task of emulating a LAN in the provider core. One property of a LAN is to maintain a single broadcast domain. That is, if a broadcast, multicast, or unknown unicast packet is received on one of the attachment circuits, it has to be sent to all other CE devices participating in that VPLS instance. The PE device handles this case by sending such a packet on all other attachment circuits and all the emulated circuits originating from that PE. With a full-mesh of emulated VCs, such a packet will reach all other PE devices in that VPLS instance. (See Figure E-10.)

*Figure E-10* ***Full Mesh of Emulated VCs***



# VPLS for an Ethernet-Based (L2) Provider Core

With an Ethernet-based provider core, customer traffic forwarding is trivial in the core. VPLS for an Ethernet-based provider core is a multipoint Layer 2 VPN that connects two or more customer devices using 802.1Q-in-Q tag-stacking technology. A VPLS essentially emulates an Ethernet switch from a users perspective. All connections are peers within the VPLS and have direct communications. The architecture is actually that of a distributed switch.

For more information on VPLS for an Ethernet-based provided core, see the following sections:

- Multipoint EWS (EP-LAN) for an Ethernet-Based Provider Core, page E-13
- Multipoint ERS (EVP-LAN) for an Ethernet-Based Provider Core, page E-13
- Topology for Ethernet-Based VPLS, page E-13

## Multipoint EWS (EP-LAN) for an Ethernet-Based Provider Core

Multipoint EWS (also known as EP-LAN in MEF terminology) is a service that emulates a point-to-point Ethernet segment. The EWS service encapsulates all frames that are received on a particular User to Network Interface (UNI) and transports these frames to a single egress UNI without reference to the contents contained within the frame. This service operation means that EWS can be used for untagged or VLAN tagged frames and that the service is transparent to all frames offered. Because the EWS service is unaware that VLAN tags might be present within the customer frames, the service employs a concept of “All to One” bundling.

## Multipoint ERS (EVP-LAN) for an Ethernet-Based Provider Core

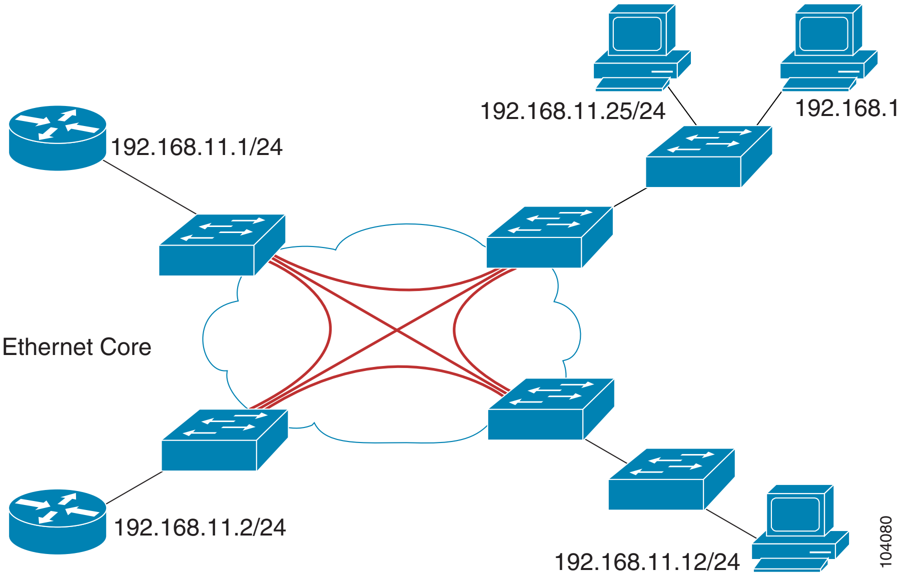
Multipoint ERS (also known as EVP-LAN in MEF terminology) models the connectivity offered by existing Frame Relay networks by using VLAN indices to identify virtual circuits between sites. ERS does, however, offer a far greater degree of QoS functionality depending upon the service provider's implementation and the customer's acceptance of VLAN indices that are administratively controlled by the service provider. Additionally, ERS service multiplexing capability offers lower cost of ownership for the enterprise as a single interface can support many virtual interfaces.

## Topology for Ethernet-Based VPLS

Ethernet-based VPLS differs from the point-to-point L2VPN definitions of EWS (EP-LAN) and ERS (EVP-LAN) by providing a multipoint connectivity model. The VPLS service does not map an interface or VLAN to a specific point-to-point pseudowire, but instead it models the operation of a virtual Ethernet switch. VPLS uses the customer’s MAC address to forward frames to the correct egress UNI within the service provider’s network for the EWS (EP-LAN).

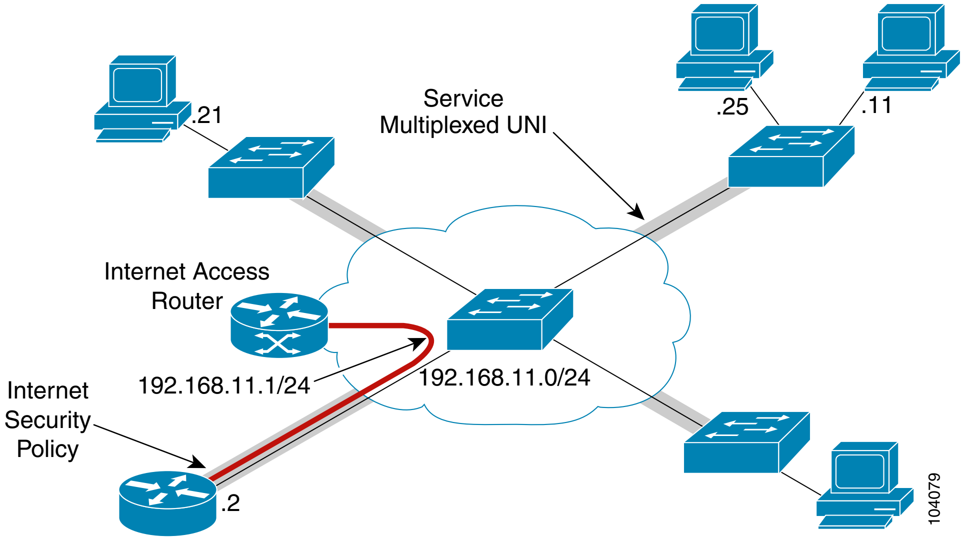
The EWS (EP-LAN) service emulates the service attributes of an Ethernet switch and learns source MAC to interface associations, flooding unknown broadcast and multicast frames. Figure E-11 illustrates an EWS (EP-LAN) VPLS topology.

*Figure E-11 VPLS EWS Topology*



The Ethernet Relay Service (ERS or EVP-LAN) offers the any-to-any connectivity characteristics of EWS and the service multiplexing. This combination enables a single UNI to support a customer's intranet connection and one or more additional EVCs for connection to outside networks, ISPs, or content providers. Figure E-12 illustrates an ERS (EVP-LAN) VPLS multipoint topology.

*Figure E-12 VPLS ERS (EVP-LAN) Multipoint Topology*



# INDEX